



Fastwel I/O изнутри

Александр Локотков

В статье рассматриваются внутреннее устройство и принципы функционирования основных составных частей аппаратно-программного комплекса Fastwel I/O, предназначенного для создания автоматизированных систем сбора данных и управления. Представлены подходы к проектированию и детально описаны межмодульная внутренняя шина FBUS, адаптированная среда исполнения прикладных программ CoDeSys, сервисы сетевых протоколов и особенности взаимодействия составных частей комплекса друг с другом.

Часть 1

О ЧЁМ ПОЙДЕТ РЕЧЬ

В данной статье рассказывается об аппаратно-программном комплексе Fastwel I/O — относительно новом семействе продукции фирмы Fastwel, предназначенном для создания автоматизированных систем сбора данных и управления технологическими процессами.

Как правило, журнальные статьи, посвящённые аппаратно-программным средствам АСУ ТП, представляют собой вариации на тему рекламных листовок или документации производителя либо являются своеобразным отчётом пользователей об опыте применения того или иного оборудования и программно-обеспечения в конкретных проектах. Ценность таких статей несомненна. В первом случае общественность получает возможность ознакомиться с основными характеристиками представляемой системы, изложенными в сжатом виде на странице-двух, а во втором — убедиться, что то или иное коммерческое «железо» или программное обеспечение используется специалистами в реальных проектах, а значит, действительно принесит кому-то пользу.

В этой статье делается попытка отступления от традиций и представляется взгляд на аппаратно-программный комплекс Fastwel I/O с противоположной точки зрения — с точки зрения разработчика. Перед началом изложения стоит сделать ряд оговорок.

Во-первых, здесь не будет подробно перечисления всех технических и функциональных характеристик изделий — при желании с ними можно оз-

накомиться в документации и справочных материалах на Web-узле компании Fastwel.

Во-вторых, автору данной статьи, являющемуся программистом, очень трудно избежать соблазна «нахваливаться» только «своё болото». Оправданием в какой-то мере может служить далеко не бесспорный тезис о главенствующей роли программного обеспечения в современных аппаратно-программных комплексах специального назначения (АСУ ТП, встраиваемые системы и т.п.). Разумеется, в материалах статьи найдется место для описания различных «тонких» приёмов прикладного программирования, конфигурирования и обслуживания комплекса.

В-третьих, здесь будут описаны не только механизмы, реализующие тот или иной набор функций, но и ход наших мыслей, инженерные компромиссы и другие факторы, лежащие в основе того или иного решения.

Насколько интересен и полезен материал такого рода, судить вам, уважаемые читатели.

Для чего изобретался «велосипед»

В состав аппаратных средств коммерческого варианта серии Fastwel I/O в настоящее время входят три типа программируемых контроллеров для сетей CANopen, RTU/ASCII и Modbus/TCP, а также более 20 типовых Modbus-модулей ввода-вывода в конструктиве WAGO-I/O-SYSTEM. Для программирования пользователям предоставляется адаптированная среда разработки программ на языках стандарта IEC 61131-3

CoDeSys (версия 2.3.6.1) фирмы 3S-Smart Software Solutions. В общем, с точки зрения конструкции и общих принципов применения, монтажа и программирования серия Fastwel I/O весьма похожа на изделия 750-й серии WAGO-I/O-SYSTEM.

Первый вопрос, который может возникнуть после прочтения приведенной информации и беглого взгляда на фотографии: а зачем, собственно, было тратить время, силы и немалые средства на разработку семейства продукции, внешне почти как две капли воды похожего на изделия фирм WAGO и Beckhoff?

Причин тому было несколько, а уж насколько они веские — решать читателю.

В начале 2003 года фирма Fastwel получила техническое задание на разработку комплекса аппаратно-программных средств для применения в составе комплексов автоматизированного управления, безопасности движения и диагностирования тягового подвижного состава (КАУД) двух типов электровозов. В частности, среди требований к контроллерам звучали такие слова, как модульность, унификация, расширенный диапазон рабочих температур, возможность разработки и встраивания прикладных алгоритмов пользователя на языке общего применения, возможность расширения номенклатуры модулей ввода-вывода без существенных временных затрат по мере развития функциональных возможностей КАУД и т.п.

В то же самое время компания Fastwel уже несколько лет выпускала продукцию в формате MicroPC, в большей степени ориентированную на OEM-производителей, и имела собственное производство.

При этом у неё не было специализированного решения для задач АСУ ТП, особенностью которого являются тяжёлые условия эксплуатации при пониженных и повышенных температурах. Следует отметить, что применение плат в формате MicroPC в системах промышленной автоматизации предполагает определённые инженерные усилия пользователя по созданию функционально и конструктивно законченного решения, что не всегда приемлемо. Одним словом, было стремление стать ближе к рынку АСУ ТП, сохранив при этом свою марку «-40...+85°C». В связи с этим было заключено соглашение с фирмой WAGO на разработку семейства продукции, по функциональным возможностям и конструкции схожего с WAGO-I/O-SYSTEM, но ориентированного на применение в широком диапазоне рабочих температур и не совместимого по внутренней межмодульной шине с модулями WAGO-I/O-SYSTEM.

Перечисленные причины требовали принятия непростого управленческого решения – либо собрать две команды, нацеленные на решение двух озвученных задач, либо попытаться реализовать оба проекта в рамках единой концепции силами одной команды. Учитывая тот факт, что у специалистов компании не было опыта создания аппаратно-программных комплексов такого класса, можно предположить, что риски, сопряжённые с выбором любого варианта, были примерно одинаковы.

Так или иначе, руководство компании выбрало второй вариант: разработчикам была поставлена задача в рамках работ над проектом КАУД разработать и реализовать единую концепцию аппаратно-программного комплекса, который сейчас носит название Fastwel I/O. При этом к разрабатываемой концепции были предъявлены следующие общие требования:

- обеспечить возможность применения в пользовательских проектах, где период исполнения прикладной программы исчисляется единицами миллисекунд, при наличии нескольких сотен каналов дискретного ввода-вывода и десятков каналов аналогового ввода-вывода;
- достичь лучшей пропускной способности, предсказуемости и универсальности межмодульной внутренней шины по сравнению с существующими аналогами;
- достичь наилучшей для изделий данного класса точности измерения аналоговых сигналов;

- максимально унифицировать основные узлы контроллеров и модулей ввода-вывода, обеспечив тем самым возможность в короткие сроки расширять номенклатуру модулей ввода-вывода;
- создать единый каркас системного программного обеспечения контроллеров, переносимый на различные операционные системы, который позволяет разработчикам наращивать набор функций, добавляя новые подсистемы по мере необходимости, а пользователям – разрабатывать прикладное программное обеспечение как на промышленных языках стандарта IEC 61131-3, так и на языке общего применения Си.

В общем, автору данной статьи повезло участвовать в этой амбициозной, рискованной и очень интересной работе, о результатах которой и пойдет речь далее.

СТРУКТУРА И ФУНКЦИИ FASTWEL I/O

Небольшое лирическое отступление. Автору категорически не нравится идея сравнения современных высокотехнологичных систем со зданиями и сооружениями, как и стремление некоторых идеологов от системотехники и инженерии упорно проводить параллели между процессом создания и реализации строительных проектов и процессом разработки сложных аппаратно-программных систем. Почему это так – тема для отдельной статьи, а может быть, и целой книги. Так или иначе, здесь не будет ни слова об архитектуре и других подобных понятиях, которые каждый «архитектор» понимает и представляет по-своему. Вместо этого разговор пойдет о структуре, которая, в понимании автора, является набором некоторых функционально обособленных составных частей и взаимосвязей между ними.

При описании какого-либо коммерческого изделия или системы обычно делается упор на его структуру, то есть на совокупность составных частей, которые потребитель может увидеть (скажем, в прайс-листе) и потрогать руками. А информация о внутреннем устройстве изделия зачастую остается за кадром – либо по причине нежелания производителя раскрывать свое ноу-хау, либо из-за убежденности в том, что потребителю такие сведения не интересны.

Отдавая дань традиции, для начала представим структуру Fastwel I/O.

Назначение Fastwel I/O

Fastwel I/O является аппаратно-программным комплексом, предназначенным для создания автоматизированных систем сбора данных и управления. Он может использоваться для построения как автономных программируемых контроллеров, так и распределённых систем сбора данных и управления. Фраза по поводу распределённых систем прежде всего относится к контроллерам CPM701 с сетевым интерфейсом CAN и протоколом прикладного уровня CANopen. Контроллеры CPM702 и CPM703, являющиеся подчинёнными узлами сетей Modbus RTU/ASCII и Modbus/TCP соответственно, также могут использоваться для реализации распределённых систем управления, роль координатора в которых играет мастер сети Modbus. Да-да, я не оговорился, уважаемые оппоненты, думающие, что распределённая система управления – это обязательно что-то вроде DeltaV с распределённым разделяемым пространством переменных.

Структура аппаратных средств

В комплекс Fastwel I/O входят следующие аппаратные средства:

- контроллеры узла сети;
- модули ввода-вывода;
- вспомогательные модули (модули питания, модули размножения потенциала, модули расширения внутренней шины и т.п.).

Контроллер узла сети является вычислительным устройством на базе микропроцессора R1610C фирмы RDC, совместимого с 80186 и имеющего тактовую частоту 100 МГц. Контроллер имеет интерфейс с модулями ввода-вывода, называемый внутренней шиной, и интерфейс внешней сети. Интерфейс внешней сети предназначен для обмена данными между контроллером и рабочими станциями, автоматизированными рабочими местами верхнего уровня автоматизированных систем сбора данных и управления, а также при использовании CANopen с другими контроллерами.

Модули ввода-вывода, подключаемые к внутренней шине контроллера, предназначены для приёма информации от датчиков и формирования управляющих воздействий на исполнительные устройства и механизмы.

Структура программного обеспечения Fastwel I/O

В комплекс Fastwel I/O входит следующее системное и инструментальное программное обеспечение (рис. 1):

- пакет адаптации среды разработки прикладных программ на языках стандарта IEC 61131-3 CoDeSys (далее – пакет адаптации CoDeSys);
- адаптированная система исполнения прикладных программ, разрабатываемых в среде CoDeSys (далее – среда исполнения CoDeSys), поставляемая в каждом контроллере узла сети;
- OPC-сервер для сетей CAN и Modbus.

Пакет адаптации CoDeSys включает в себя:

- интегрированную среду разработки IDE CoDeSys версии 2.3.6.1 фирмы 3S-Smart Software Solutions;
- файлы описания платформы Fastwel I/O, интегрируемые с IDE CoDeSys и позволяющие генерировать исполняемый код прикладных программ для контроллеров Fastwel I/O средствами IDE CoDeSys;
- файлы описания конфигурации модулей ввода-вывода, интегрируемые с IDE CoDeSys и позволяющие генерировать конфигурационную информацию для контроллеров средствами IDE CoDeSys;
- драйверы коммуникационного сервера CoDeSys Gateway Server, интегрируемые с CoDeSys Gateway Server и позволяющие выполнять загрузку прикладных программ в контроллер, удалённую отладку и мониторинг переменных из среды разработки.

Здесь стоит отметить, что пользователю для получения возможности работы с Fastwel I/O достаточно установить пакет адаптации. Если на инструментальной машине отсутствует или имеется более старая, чем 2.3.6.1, версия IDE CoDeSys, в процессе инсталляции будет предложено установить 2.3.6.1. Если на инструментальной машине установлена версия IDE CoDeSys 2.3.6.2 или 2.3.6.3, то, к сожалению, придется сначала её удалить. О причине будет рассказано несколько позднее при описании сервиса исполнения прикладных программ Fastwel I/O, в котором для выполнения пользовательских программ используется система исполнения CoDeSys для процессоров 80186.

Кроме того, следует добавить, что на момент написания данной статьи фирмой ПРОСОФТ не бы-

ла определена схема распространения OPC-серверов для сетей Modbus и CAN. Так или иначе, на компакт-диске пакета адаптации пользователь найдёт установочные комплекты демонстрационных версий OPC-серверов для сетей CAN и Modbus RTU/ASCII/TCP, а также полную бесплатную версию OPC-сервера для Modbus RTU/ASCII.

Как это использовать

Для того чтобы контроллер Fastwel I/O начал делать что-то полезное, нужно:

- 1) подключить к нему модули ввода-вывода и источник питания с выходным напряжением от 10 до 30 В постоянного тока;
- 2) в среде разработки CoDeSys создать проект для целевой платформы Fastwel I/O System;
- 3) в секции конфигурации контроллера PLC Configuration среды разработки CoDeSys выбрать тип используемого контроллера, добавить в конфигурацию описания модулей ввода-вывода, которые подключены к контроллеру, и установить требуемые значения параметров для контроллера и модулей;
- 4) если контроллер будет подключен к сети (CAN, Modbus RTU/ASCII или Modbus/TCP), добавить в конфигурацию контроллера описания коммуникационных объектов внешней сети и настроить их параметры;
- 5) в редакторе программ CoDeSys разработать программу для контроллера, которая делает то полезное, ради чего приобретались контроллер и модули ввода-вывода. Например, если нужно только управлять каналами модулей ввода-вывода по сети, к которой подключён контроллер, программе достаточно передавать дан-

ные между сетью и модулями ввода-вывода;

- б) подключить компьютер, на котором разработан проект для контроллера, к сети, к которой подключён контроллер, либо воспользоваться поставляемым кабелем для установления связи между инструментальным компьютером и контроллером через последовательные порты;
- 7) скомпилировать проект и загрузить его в контроллер (не забыв предварительно включить блок питания контроллера!), воспользовавшись соответствующим сервисом среды разработки CoDeSys.

Подробная информация о том, как выполнять перечисленные действия (за исключением пункта 5), приведена в руководстве программиста для каждого контроллера. О том, как разрабатывать программы в среде CoDeSys, рассказано в документации на CoDeSys, входящей в установочный комплект нашей адаптации CoDeSys. Ну, а о том, как разрабатывать полезные программы независимо от языка программирования и среды исполнения, написано много хороших (но еще больше – не очень) книг.

А что внутри?

В предыдущем разделе речь шла о структуре Fastwel I/O, осязаемой извне. Поговорим о внутренней структуре аппаратно-программных средств Fastwel I/O и о факторах, повлиявших на принятие тех или иных проектных решений.

Что должен делать программируемый логический контроллер (ПЛК), к тому же являющийся узлом сети передачи данных? Во время функционирования некоторой системы управления, частью которой является ПЛК, контроллер ис-

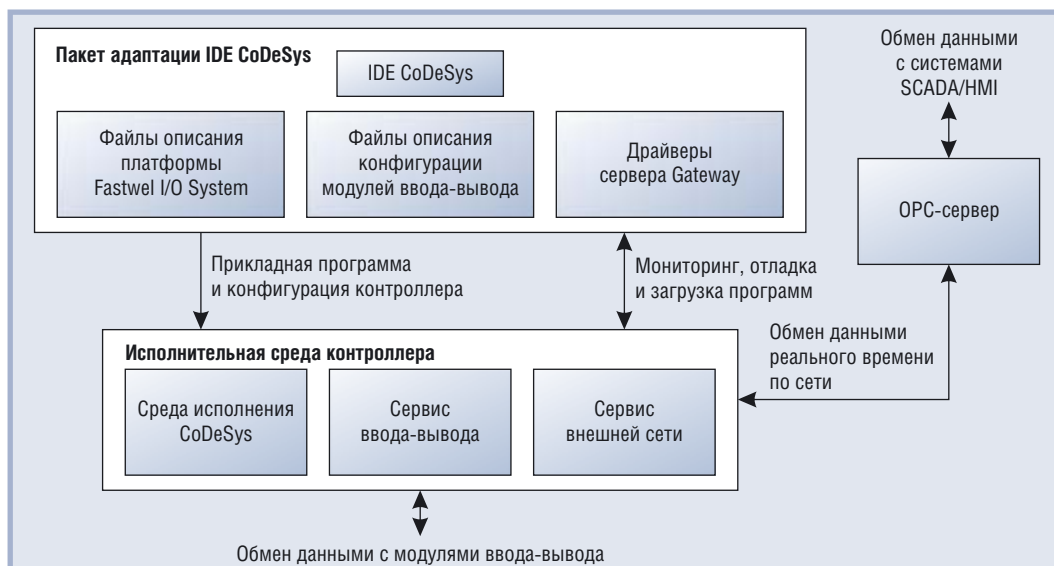


Рис. 1. Структура программного обеспечения Fastwel I/O



Рис. 2. Циклограмма исполнения прикладной программы

полняет прикладную управляющую программу, разработанную пользователем, обменивается данными с модулями ввода-вывода, обеспечивая взаимодействие прикладной программы с контролируемым объектом, а также осуществляет информационный обмен по сети, к которой он подключён. Взаимодействие прикладной программы контроллера с контролируемым объектом и другими узлами сети будет далее называться взаимодействием с окружением.

Традиционно исполнение прикладной программы ПЛК циклично и полностью синхронно относительно операций обмена данными с окружением. То есть во время очередного цикла программы данные, полученные программой из окружения, не могут (и не должны!) неожиданно измениться до окончания текущего цикла. Кроме того, обновление данных, выводимых программой в окружение, происходит точно в конце очередного цикла.

С учётом указанных особенностей модель окружения прикладной программы обычно представляется так называемым образом процесса, который состоит из двух непересекающихся областей памяти. Первая область образа процесса, на-

зываемая *областью входных данных*, предназначена для хранения и обновления значений входных данных прикладной программы в процессе приёма информации от устройств ввода-вывода и сетевых интерфейсов. Вторая область называется *областью выходных данных* и служит для хранения и обновления значений выходных данных программы в процессе выдачи информации устройствам ввода-вывода и в сетевые интерфейсы. Упрощённый вариант циклограммы функционирования контроллера показан на рис. 2.

Здесь стоит вспомнить, что перед началом нашего проекта ставилась задача достижения миллисекундных периодов исполнения. Глядя на упрощённую циклограмму ПЛК, можно сделать вывод, что наиболее важными факторами в достижении заданного показателя производительности являются производительность центрального процессора и пропускная способность межмодульной внутренней шины.

Качество кода, генерируемого компилятором прикладных программ, пока выносится за скобки.

Предположим, что к внутренней шине контроллера подключены модули с 512

каналами дискретного ввода-вывода длиной 1 бит и с 50 16-разрядными каналами аналогового ввода-вывода. Простой расчёт позволяет сделать вывод, что для чтения и записи такого количества каналов за 1 мс пропускная способность внутренней шины с учётом всех накладных расходов должна быть не хуже 164 кбайт/с.

Далее полагаем, что прикладная программа должна в каждом цикле выполнять следующие операции с плавающей точкой (с операндами одинарной точности) над значением каждого канала аналогового ввода-вывода: одно преобразование целого двухбайтового значения, одно умножение, одно сложение или вычитание и одно сравнение. Таким образом, для того чтобы успевать проделывать перечисленные нехитрые операции со значениями 50 аналоговых каналов, производительность процессора должна быть не хуже 200000 FLOPs (операций с плавающей точкой в секунду). Разумеется, это довольно грубая оценка, поскольку в реальной жизни операций с плавающей точкой в программе может быть значительно больше или не быть вовсе. Кроме того, перечисленные операции с плавающей точкой требуют разного количества машинных тактов. Наконец, прикладная программа в ПЛК не одинока — процессор может потребоваться другим подсистемам.

Так или иначе, при отсутствии чётко сформулированных критериев проектирования ключевых компонентов системы подобные инженерные оценки имеют право на жизнь.

Структура контроллера

Структурная схема контроллера Fastwel I/O представлена на рис. 3. При

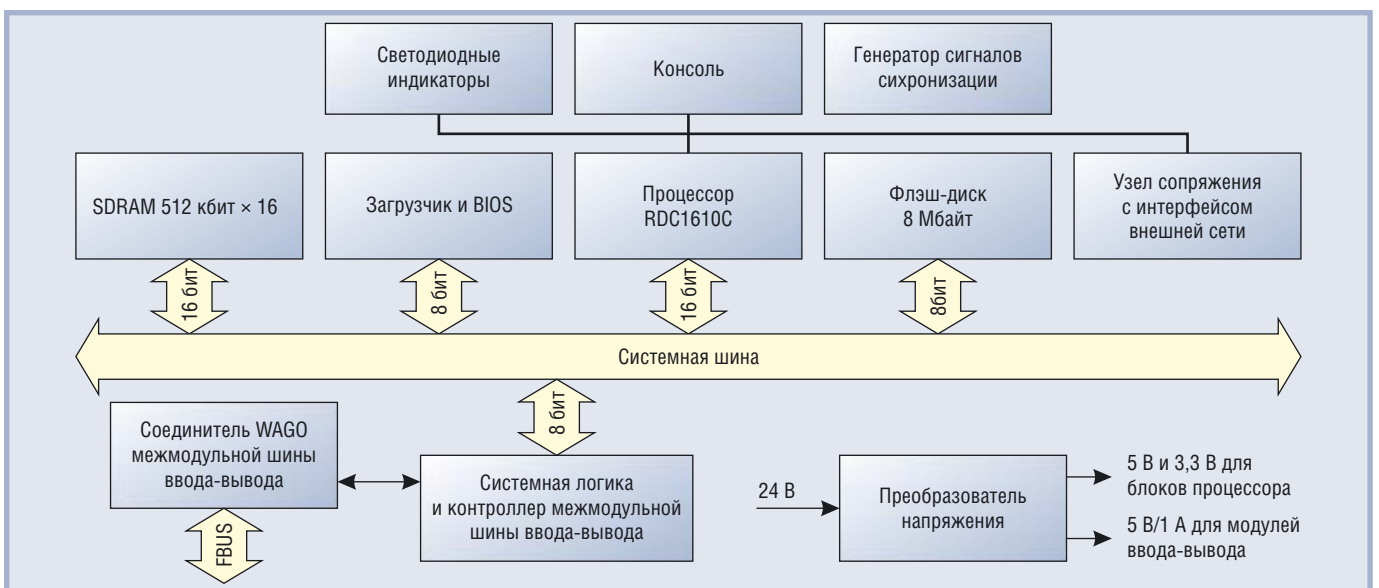


Рис. 3. Структурная схема контроллера CPM70x

включении питания контроллера первым делом происходит загрузка операционной системы Fastwel DOS, а значит, пользователь имеет дело с ПЛК на базе IBM PC совместимого вычислительного устройства.

Спрашивается, зачем внутри ПЛК, программируемого на CoDeSys, иметь совместимый с 80186 процессор, BIOS и DOS? Причин тому несколько:

- исторически компания Fastwel производила только IBM PC совместимые вычислители, а значит, у нас имелись наработки как в части схемотехнических решений, так и в области системного программного обеспечения. Переход на другую платформу был бы сопряжён со значительными вложениями и временными затратами на освоение новой платформы специалистами или потребовал бы привлечения к работе большого количества новых специалистов;
- как указано ранее, в качестве одного из основных требований выдвигалась возможность разработки прикладных программ на языках общего применения типа Си. В этой связи мы подумали, что в силу гораздо большей распространённости IBM PC совместимых компьютеров среди наших потенциальных пользователей имеется очень много разработчиков на языках программирования общего применения, владеющих популярными средствами разработки для IBM PC от Borland и др. На момент начала разработки не было уверенности в качестве компилятора GCC для встраиваемых 32-разрядных процессоров. К слову, её не было, потому что никто тогда у нас этой темой не занимался;
- многим пользователям ПЛК, разрабатывающим программы на проблемно-ориентированных языках IEC 61131-3, по большому счёту, абсолютно безразлично, какой процессор и операционная система установлены в их ПЛК. В то же время в момент начала разработки появился комплект адаптации пакета CoDeSys для процессоров, совместимых с 80186, причём с весьма умеренной стоимостью лицензии среды исполнения, устанавливаемой в одно вычислительное устройство;
- у нас имелся практически полный набор инструментальных средств и библиотек для IBM PC совместимых вычислителей, включая комплект адаптации BIOS, операционную сис-

тему Fastwel DOS, MS Visual C++, Borland C++ 3.1, мультизадачное ядро CMX-RTX, стек CMX-TCP/IP и т.п., а также опыт применения перечисленных средств.

Процессор R1610C на момент начала разработки казался нам наиболее производительным и интегрированным решением за счет следующих факторов:

- при тактовой частоте 100 МГц процессор имеет систему команд 80186, причём каждая команда выполняется за меньшее количество тактов, чем у ближайших аналогов Intel и AMD;
- на кристалле имеется кэш-память инструкций и данных по 8 кбайт на каждую область;
- наличие интерфейса с синхронной динамической памятью;
- наличие двух встроенных адаптеров сети Ethernet.

Учитывая весьма умеренный ток потребления и привлекательную цену за один такой кристалл, R1610C показался нам идеальным для решения поставленной задачи.

Остальные элементы структуры вычислительного ядра достаточно традиционны для IBM PC совместимых микрокомпьютеров, за исключением блока приёмопередатчика межмодульной внутренней шины, реализованного на базе программируемой логической интегральной схемы Xilinx, часть которой также используется для выполнения различных функций системной логики. Особенности построения межмодульной внутренней шины рассматриваются далее.

В результате у нас получилось конструктивно законченное универсальное вычислительное ядро, устанавливаемое в пластиковый корпус WAGO, и отдельные модули интерфейсов внешней сети, присоединяемые к плате вычислительного ядра под углом 90° и размещаемые в том же корпусе.

Структура модуля ввода-вывода

Обобщённая структура модуля ввода-вывода представлена на рис. 4.

Применение микроконтроллера в каждом модуле было обусловлено следующими факторами.

1. Согласно требованиям к модулям ввода-вывода для проекта КАУД, количество входных каналов у одного модуля достигало 16, а выходных — до 24, причём речь шла не только о дискретных каналах, значения которых укладываются в один

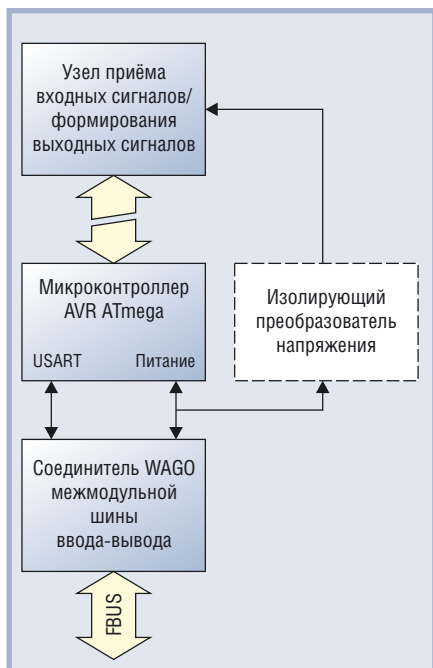


Рис. 4. Обобщённая структурная схема модуля ввода-вывода

бит. Например, в одном специализированном модуле дискретного управления в настоящее время имеются 16 физических дискретных входов; 8 каналов измерения тока нагрузки, протекающего по каждому выходному ключу повышенной мощности; 8 каналов, представляющих реальное состояние каждого из 8 выходных ключей; 8 каналов дискретного вывода для управления выходными ключами; канал управления реле подключения нагрузок к выходам и несколько вспомогательных каналов. Разумеется, модули с таким количеством каналов имеют отличную от WAGO конструкцию, поскольку у WAGO тогда было не более 8 контактов на модуль, и очень малые габариты для рассеяния довольно приличных мощностей выходных каналов модулей дискретного управления. Представление о конструкции специализированных

модулей можно получить, посмотрев на рис. 5.

2. Высокоточное измерение аналоговых сигналов требует применения соответствующих аналого-цифровых преобразователей, управление и обмен данными с которыми обычно выполняются микроконтроллерами по последовательному каналу типа SPI. Такой вариант подключения АЦП позволяет реализовать довольно экономичное решение по гальванической развязке каналов ввода-вывода от межмодульной шины, к которой подключается модуль.
3. При производстве модулей аналогового ввода-вывода с высокими показателями точности требуется проводить специальные процедуры калибровки, по окончании которых калибровочные коэффициенты должны сохраняться в энергонезависимой памяти.
4. Планировалась разработка модулей последовательных интерфейсов типа RS-232C и RS-485. Без микропроцессора на плате сделать хорошо такие модули весьма непросто.
5. Однозначно не планировалось ограничиваться только конструктивом WAGO.

Наконец, перед началом разработки мы имели довольно приблизительное представление о том, как должен выглядеть протокол обмена по межмодульной шине, а значит, не могли в точности оценить требования к вычислительным ресурсам для его реализации. Что мы знали точно, так это скорость обмена по последовательной межмодульной шине, к которой нужно стремиться: 2 Мбит/с. А это означает ворох прерываний, обрушивающийся на «голову» модуля, в худшем случае, примерно один раз в 5 мкс, при том что ещё нужно что-то измерять, вычислять и заниматься другими полезными делами.

В общем, изложенные факторы привели нас к решению использовать в модулях ввода-вывода микроконтроллеры AVR серии ATmega при тактовой частоте 16 МГц. При таком значении тактовой частоты один из встроенных USART микроконтроллера мог быть настроен как раз на требуемые 2 Мбит/с. ●

**Автор — сотрудник фирмы Fastwel 119313, Москва, а/я 242
Тел.: +7 (495) 234-0639
Факс: +7 (495) 232-1654
E-mail: info@fastwel.ru
Web: www.fastwel.ru**



Рис. 5. Внешний вид модульного контроллера и специализированных модулей ввода-вывода для системы ЕКС-2 электровоза ЧС7