

Краткий экскурс в историю промышленных сетей

В статье на примере популярных стандартных решений рассмотрена проблема практического применения промышленных сетевых технологий. Хочется сразу предупредить читателя о том, что статья не претендует на то, чтобы стать обзором всех доступных промышленных сетей. Такие обзоры регулярно публикуются практически всеми изданиями, имеющими отношение к промышленной автоматизации или компьютерному рынку. В данной работе автор попытался провести анализ наиболее часто применяемых в современных условиях решений. Возможно, некоторые из них вызовут интерес специалистов отделов АСУП и системных интеграторов, стоящих перед проблемой выбора направления дальнейшего развития своих систем.

Сергей Гусев

root@prosoft.ru

ПОЧЕМУ?

Рискуя повторить в сотый раз банальную истину, все-таки приведем три основные предпосылки, создающие поистине революционную ситуацию, которая обуславливает сегодня повсеместное применение распределенных сетевых технологий разработчиками систем АСУ.

1. Изделия из кремния дешевеют, изделия из меди дорожают.

В последние годы эта тенденция стала особенно заметна. Прошли те времена, когда нормой жизни считался огромный шкаф, напичканный автоматикой, с выходящими из него толстыми пучками кабелей, ведущими к датчикам и исполнительным механизмам. Сегодня в большинстве случаев становится экономически целесообразной установка на площади цеха или участка нескольких локальных контроллеров или интеллектуальных УСО, объединенных в единую сеть, чем прокладка разветвленных кабельных систем.

2. Стоимость работ по установке, тестированию, вводу в эксплуатацию и сопровождению централизованной системы гораздо выше, чем распределенной.

Количество проводных соединений в централизованной системе как минимум в два раза больше, чем в распределенной (рис. 1). Нужно учитывать многократно возрастающую вероятность ошибки при монтаже проводников в многочисленных кроссовых клеммных колодках и сложность поиска и устранения неисправностей. Отдельно стоит упомянуть о ситуации, когда в составе объекта управления появляется еще несколько входных или выходных ка-

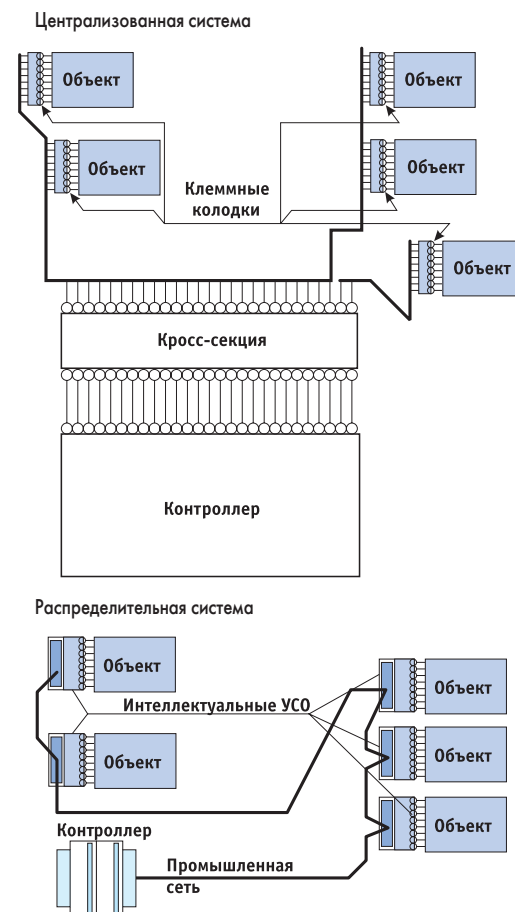


Рис. 1. Принципы построения централизованных и распределенных сетей

налов. Добавление новых линий связи к уже проложенной кабельной системе — занятие не из простых.

3. Растущая потребность в «распределенном интеллекте».

Сегодня, когда микропроцессоры и другие специализированные микросхемы достаточно подешевели, стало целесообразным выделять в общей системе АСУ отдельные локальные задачи и решение их поручать локальным контроллерам. Контур управления, таким образом, замыкается на нижнем уровне. Сеть же позволяет контроллерам в качестве аргументов для вычисления управляющего вектора использовать переменные других контроллеров, обеспечивая связанность системы управления в целом. Такая архитектура существенно увеличивает производительность, надежность и масштабируемость систем. Кроме того, современные исполнительные механизмы, как правило, сами являются интеллектуальными и законченными «субъектами» промышленных сетей.

ЧТО ТАКОЕ FIELDBUS?

Так пишется оригинальный термин, который в русском переводе звучит как «промышленная сеть». Fieldbus — это не какой-то определенный протокол передачи данных и не тип сетевой архитектуры. Этот термин не принадлежит ни одной отдельно взятой компании и обозначает скорее сферу применения, чем какую-либо конкретную сетевую технологию. Промышленные сети — это сегмент рынка, где сталкиваются интересы крупнейших корпораций, создаются и внедряются самые передовые технологии, ведется война стандартов, появляются новые участники и стараются удержаться старые. Промышленная сеть — это среда передачи данных, которая должна отвечать множеству разнообразных, зачастую противоречивых требований. Промышленная сеть — это набор стандартных протоколов обмена данными, позволяющих связать воедино оборудование различных производителей, а также обеспечить взаимодействие нижнего и верхнего уровней АСУ. Наконец, промышленная сеть — это образ мысли инженера, определяющий конфигурацию и принципы построения системы. От того, какая сетевая архитектура выбирается сегодня, будут зависеть не только затраты на создание системы, но и срок ее жизни, ее способность к развитию, то есть, как принято сейчас говорить, интегральная стоимость владения.

Сформулируем лишь некоторые основные требования, которые можно предъявить к «идеальной» промышленной сети.

1. Производительность.
2. Предсказуемость времени доставки информации.
3. Помехоустойчивость.
4. Доступность и простота организации физического канала передачи данных.
5. Максимально широкий сервис для приложений верхнего уровня.

6. Минимальная стоимость устройств аппаратной реализации, особенно на уровне контроллеров.

7. Возможность получения «распределенного интеллекта» путем предоставления нескольким ведущим узлам максимального доступа к каналу.

8. Управляемость и самовосстановление в случае возникновения нештатных ситуаций.

Как видно, в получившемся списке первое требование противоречит второму, третье — четвертому и т. д. Более того, подобных противоречий приходится избегать постоянно и на всех уровнях проектирования, начиная с того, какой формат пакета передачи данных выбрать: тот, который позволит осуществлять расширенное управление сетью и удаленную загрузку, или тот, который обеспечит максимально быструю работу с большим числом дискретных сигналов, заканчивая решением философской проблемы, что лучше: применить не самое современное, но проверенное годами решение или применить кажущееся блестящим и современным решение, которое почему-то оказывается дороже и еще до сих пор не применяется на предприятии-конкуренте.

Таким образом, можно полагать, что промышленная сеть — это один большой компромисс. И от того, как расставлены акценты в этом компромиссе, зависит успешность решения задач, стоящих перед сетевой архитектурой. К промышленным сетям вполне применимы результаты теоретических изысканий в области коммуникационных сетей общего назначения, поэтому становятся вполне понятными постоянные ссылки на «Модель взаимодействия открытых систем» (Open System Interconnection model), принятую Международной организацией по стандартизации (ISO). Уместно напомнить назначение всех семи уровней этой модели. Попробуем кратко проанализировать приведенные далее сетевые решения именно с этой точки зрения. А именно, постараемся понять, для решения каких основных задач создавались эти сети, чем разработчики жертвовали, что выдвигали на первый план, и какие возможности это дает конечным пользователям.

ЭКСКУРС В ТЕОРИЮ

Предлагаю выделить три наиболее значимых параметра, по которым можно некоторым образом сравнивать сети, и которые послужат основой для дальнейшей классификации, а именно: топология сети, объем информационного сервиса, предоставляемого сетью, и способ доступа к физическому каналу передачи данных.

Топология сети

Наиболее распространенный тип сетевой топологии — это общая шина. Основное ее преимущество заключается в простоте, дешевизне и легкости переконфигурирования. Такая сеть не боится отключения или подключения устройств во время работы. Хорошо под-

ходит для сильно распределенных объектов. Имеет ряд «генетических» недостатков: присутствие в каждой точке сети общего трафика, опасность потери связи при одиночном обрыве канала связи или фатальном выходе из строя одного узла.

Топология типа «кольцо» очень популярна со времен выхода на рынок сети Token Ring фирмы IBM. Использование протокола с циклической передачей маркера (IEEE-802.5) позволяет сетям с такой топологией обеспечить абсолютную предсказуемость и хорошую пропускную способность. Основными недостатками топологии являются высокая стоимость организации канала связи, нерациональное (в большинстве случаев) использование сетевого трафика и потеря всей синхронизации сети в случае сбоя и отключения хотя бы одного из узлов.

Топология «звезда», являясь логическим продолжением моноканала, обеспечивает дополнительную защиту всей сети от выхода из строя или отключения узлов, позволяет существенно оптимизировать трафик, передавая пакеты только в те «лучи», где находятся их получатели. Последнее особенно существенно для сетей, где допускаются коллизии.

Объем информационного сервиса

Существует хорошо известная модель Международной организации по стандартизации (ISO/OSI), призванная разграничить и формализовать функции, выполняемые различными уровнями аппаратного и программного обеспечения сетевой структуры. Данная модель определяет семь уровней сервиса, предоставляемого сетью (табл. 1).

Таблица 1. Уровни, определяемые моделью взаимодействия открытых систем (ISO-Open System Interconnection Model)

7	Application	Прикладной уровень
6	Presentation	Уровень представления
5	Session	Уровень сессий
4	Transport	Транспортный уровень
3	Network	Сетевой уровень
2	Data Link	Канальный уровень
1	Physical	Физический уровень

- На физическом уровне определяются физические характеристики канала связи и параметры сигналов.
- Канальный уровень формирует основную единицу передаваемых данных — пакет — и отвечает за дисциплину доступа устройства к каналу связи (Medium Access Control) и установление логического соединения (Logical Link Control).
- Сетевой уровень отвечает за адресацию и доставку пакета по оптимальному маршруту.
- Транспортный уровень разбирается с содержанием пакетов, формирует ответы на запросы или организует запросы, необходимые для уровня сессий.
- Уровень сессий оперирует сообщениями и координирует взаимодействие между участниками сети.

- Уровень представления занимается преобразованием форматов данных, если это необходимо.
- Прикладной уровень — это набор интерфейсов, доступных программе пользователя.

На практике большинство промышленных сетей ограничивается только тремя из них, а именно физическим, канальным и прикладным. Наиболее «продвинутые» сети решают основную часть задач аппаратно, оставляя программную прослойку только на седьмом уровне. Дешевые сети (например, ModBus) зачастую используют на физическом уровне RS-232 или RS-485, а все остальные задачи, начиная с канального уровня, решают программным путем.

Тип доступа к физическому каналу

Говоря языком ISO/OSI, это второй, канальный, уровень модели. На самом деле, по большому счету существуют два типа доступа: с коллизиями и без. Доступ к каналу с коллизиями используют Ethernet, CAN и LON. Такой тип доступа позволяет эффективно использовать пропускную способность канала и предоставлять доступ в сеть нескольким активным узлам.

Единственным недостатком такого подхода являются собственно коллизии, которые не позволяют указанным сетям на равных конкурировать с детерминированными протоколами в ряде задач. Для разрешения коллизий применяются различные приемы.

Например, в сетях Ethernet применяется технология CSMA/CD (Carrier Sense Multiple Access with Collision Detection). Технология основана на постоянном прослушивании линии всеми узлами и генерации повторной попытки занятия канала через случайный промежуток времени в случае, если обнаружена попытка одновременного доступа к каналу нескольких станций.

Принципиально другую форму разрешения коллизий демонстрирует CAN. Его протокол относится к классу CSMA/CR (Carrier Sense Multiple Access with Collision Resolution). Разрешение коллизий производится аппаратурой по принципу побитового сравнения сетевых адресов конфликтующих устройств (рис. 2). Станция, пытающаяся передать очередную «единицу» из своего адреса, видя, что реально в канале передается «ноль», понимает, что конфликтует, и откладывает попытку

```

10011011011 — Передача 1 станции
10011011 1... — Передача 2 станции
                момент прекращения
10011011 011 — Результат передачи
  
```

Рис. 2. Разрешение коллизий по CAN-протоколу

занять канал до лучших времен. Станция, передающая «ноль», спокойно продолжает свое дело. Таким образом, хотя коллизии и возникают, но разрешаются предсказуемо и в предсказуемое время. Именно это позволило сетям на основе протокола CAN занять достойное место в различных отраслях, особенно в авто-

мобилестроении, где важны мультимастерные сети с распределенным интеллектом.

Дальнейшее развитие данная технология получила в сетях LON. Только в отличие от CAN аргументами в споре за канал являются не сетевые адреса, а динамически изменяемые приоритетные уровни пакетов, что позволяет, например, пакету, несущему важную информацию и требующему немедленного ответа, легко «пробиться» через поток низкоприоритетных информационных обменов.

Однако основная масса промышленных сетевых протоколов использует детерминированный способ доступа к каналу по принципу «запрос-ответ» или с помощью передачи маркера. Это эффективный путь для организации четкого и ритмичного сетевого взаимодействия. В основе протоколов с передачей маркера лежит принцип постоянного наличия в сети синхронизирующего пакета, называемого маркером (рис. 3).

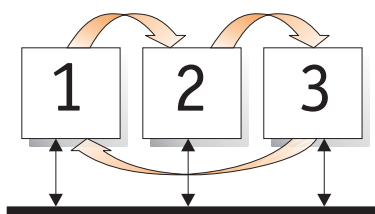


Рис. 3. Последовательность прохождения маркера среди активных участников сети

Перейдем к краткому рассказу о нескольких конкретных сетевых решениях, представляющих, с точки зрения автора, наибольший интерес по популярности на нашем рынке в ближайшее время.

FOUNDATION FIELDBUS

Несмотря на отсутствие широкой практики внедрения этого стандарта в России, хотелось бы остановиться на данном решении, так как подкупают его необыкновенная красота и продуманность. Foundation Fieldbus (далее FF) — самый молодой и быстро растущий стандарт на промышленную сеть. Он вобрал в себя современные технологии построения управляющей сети масштаба предприятия.

FF представляет собой двухуровневый сетевой протокол, сочетающий черты мощной информационной магистрали для объединения компьютеров верхнего уровня и управляющей сети, объединяющей контроллеры, управляющие компьютеры, датчики и исполнительные механизмы. Предоставляет полный сервис, от передачи файлов и больших объемов информации до замыкания контуров управления контроллеров, включая обеспечение загрузки в контроллеры управляющих программ и доступ к пассивному оборудованию. Все это находится в рамках одного стандарта.

Звучит просто фантастически. Еще невероятнее выглядят прогнозы на ближайшее буду-

щее. По оценкам Ассоциации интеграторов управляющих систем (Control System Integrators Association — CSIA), в 2001 году не менее 80 % вновь создаваемых систем будет совместимо со стандартом FF в части сетевых технологий.

Практически стандарт определяет два уровня сети. На нижнем уровне (Н1) в качестве физической среды передачи данных за основу взят стандарт IEC 61158-2, который позволяет использовать сеть FF на взрывоопасных производствах с возможностью запитки датчиков непосредственно от канала связи.

Скорость передачи информации на уровне Н1 составляет 31,5 Кбит/с.

На верхнем уровне (бывший Н2) в настоящее время, как правило, используется FF HSE (High Speed Ethernet), основанный, как видно из названия, на сети Ethernet со скоростью 100 Мбит/с.

Особенностью стандарта FF является то, что в нем определен дополнительный пользовательский уровень (User Layer), позволяющий, применяя predetermined функциональные блоки, строить промышленные сети с распределенным интеллектом.

PROFIBUS

В развернутом представлении этот стандарт не нуждается. Он весьма популярен в Европе и особенно в Германии. Активно продвигается в качестве стандартного решения компанией Siemens. Представляет собой классическую сеть на базе общей шины с передачей маркера.

PROFIBUS существует в трех основных вариантах:

PROFIBUS-DP — быстрый (до 12 Мбит/с) одномастерный протокол. Физическая среда передачи — экранированная витая пара стандарта RS-485. Хорошо подходит для построения быстрых детерминированных распределенных систем сбора данных и управления с одним ведущим узлом.

PROFIBUS-FMS включает в себя дополнительные типы пакетов (Fieldbus Message Specification). Позволяет организовывать в одной сети работу нескольких активных станций.

PROFIBUS-PA — сетевой интерфейс, физическая среда передачи данных которого соответствует требованиям стандарта IEC 61158-2. Может применяться для построения сети, соединяющей исполнительные устройства, датчики и контроллеры, расположенные непосредственно во взрывоопасной зоне.

На прикладном и канальном уровнях PROFIBUS-PA использует весь сервис, доступный в PROFIBUS-FMS. На физическом уровне интерфейсы Н1 Foundation Fieldbus и PROFIBUS-PA используют одинаковую витую пару, одинаковые уровни сигналов и скорости передачи и позволяют окончательным устройствам запрашиваться непосредственно от канала связи.

Более того, два этих протокола могут одновременно ужиться на одном и том же физическом участке сети. Просто канальный уровень каждого из протоколов «не понимает» пакеты конкурента.

Какие же задачи наиболее просто решаются с помощью PROFIBUS?

Во-первых, это модернизация и расширение возможностей существующих систем. Предположим, что вы уже имеете Simatic S7, в который можно добавить только один модуль расширения, а количество каналов, которые необходимо добавить в систему, превышают максимально возможное для данного типа модуля. После несложных расчетов вы поймете, что лучшим решением в данном случае будет приобретение мастер-карты PROFIBUS-DP для вашего S7 и интеллектуальных распределенных УСО типа WAGO I/O.

С точки зрения программирования, вы не заметите разницы между данными, получаемыми PLC из локального модуля, и информацией от удаленных УСО. Более того, если вообще не устанавливать в S7 никакие модули, кроме сетевой карты, а весь ввод/вывод реализовать на распределенных УСО WAGO I/O, вы получите не только существенный «экономический эффект», но и более гибкую, распределенную и легко масштабируемую систему. Именно по этому пути сегодня производится модернизация «классических», построенных на традиционных PLC европейских производств.

Следующая задача, которую часто приходится решать разработчику, — это создание «с чистого листа» новой распределенной системы сбора данных и управления.

Выбор PROFIBUS в качестве сетевой среды сегодня выглядит вполне оправданным. Столь же оправданным оказывается выбор IBM PC-совместимых контроллеров в качестве основных управляющих узлов сети. Возможны несколько приемлемых конфигураций, выбор которых определяется поставленной задачей.

Если требуется объединить в детерминированную сеть несколько контроллеров, оптимальным вариантом будет PROFIBUS-FMS. Для создания сети с централизованным интеллектом и распределенным вводом/выводом лучше всего подойдет PROFIBUS-DP.

Наиболее простой способ построения системы показан на рис. 4. В этом случае цикл

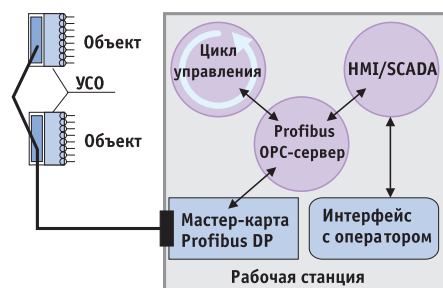


Рис. 4. Централизованное управление, распределенные УСО

управления замыкается внутри рабочей станции, которая выступает одновременно в роли операторской станции и программного аналога PLC. Для этого в ней устанавливается мастер-карта PROFIBUS-DP, а ведомые (slave) узлы, такие как WAGO I/O или ET200, подключаются к ней по топологии «общая шина».

Логически потоки данных в такой сети делятся на три основных цикла.

1. Цикл ввода/вывода выполняется под управлением контроллера ведомого узла. В этом цикле происходит автоматический опрос модулей ввода, установленных в УСО, и строится таблица последних значений, готовых к передаче в сеть. Одновременно с этим происходит передача выходным модулям УСО новых значений, полученных из сети. Длительность этого цикла зависит от количества установленных модулей и, как правило, измеряется единицами миллисекунд.

2. Цикл сетевого обмена реализуется по инициативе ведущего узла, в данном случае по маркеру мастер-карты рабочей станции. В этом цикле ведущий формирует пакеты, содержащие данные для модулей вывода каждого из абонентов, и принимает от них пакеты, в которых передается информация от входных модулей. Пакеты оптимизированы таким образом, что на передачу данных отводится ровно столько места, сколько эти данные занимают. Например, передача аналогового сигнала занимает в сетевом пакете два байта, а передача дискретного — один бит. Служебная информация в пакетах предельно мала, поэтому теоретическая пропускная способность сети уменьшается в основном только в связи с издержками на передачу и обработку маркера, а также из-за того, что параметры передаются по сети независимо от того, изменилось их значение со времени предыдущего цикла опроса или нет. Цикл сетевого обмена осуществляется без участия центрального процессора рабочей станции и начинается сразу после подачи напряжения питания на сетевую карту и сетевые УСО. Данные, которые передаются ведомым абонентам, ведущий постоянно берет из определенного поля адресов специальной двухпортовой памяти. В эту же память после каждого цикла обмена по сети помещаются новые значения, полученные от каналов ввода. Для ускорения сетевого обмена данные в пакетах передаются подряд, без указания их источника или, наоборот, адресата. Для того чтобы ведущий «знал», какие из участков своей двухпортовой памяти передать каждому из абонентов и, соответственно, какую длину пакета ожидать в ответ и что это будет обозначать, такая сеть изначально должна быть однократно сконфигурирована с помощью специальной программы. В результате работы программа-конфигуратор настраивает ведущего и ведомых участников друг на друга и сохраняет информацию о параметрах сети в энергонезависимой памяти всех узлов. Такая дисциплина работы PROFIBUS-DP, конечно, не допускает «горячего» (на ходу) изменения числа участников сети и даже состава их модулей ввода/вывода, но зато обеспечивает высокую скорость обмена. Так, например, цикл обмена по сети, которая обслуживает 5000 дискретных сигналов и 1000 аналоговых, может составлять менее 2 мс!

3. Цикл управления внутри рабочей станции.

Эта работа возлагается на центральный процессор. Он работает с так называемым образом процесса, который находится в двухпортовой памяти сетевой карты. Процессору требуется считать из памяти информацию о входных каналах, осуществить над ней необходимые преобразования и выдать управляющие воздействия, занеся в определенные ячейки памяти новые данные.

Такая конфигурация управляющей системы по принципу работы и программирования почти ничем не отличается от «вырожденной» централизованной системы. Один и тот же процессор здесь отвечает и за управление, и за интерфейс с оператором. Преимущества, которые получены на данном этапе, — это освобождение процессора от задач ввода/вывода (обслуживание прерываний от АЦП, поддержка каналов DMA, необходимость работы с резидентными драйверами устройств и т. п.), а также возможность максимально приблизить УСО к объекту контроля.

Однако для многих задач такой подход не обеспечивает управление в реальном времени. Связано это с тем, что современное программное обеспечение операторского интерфейса в своей массе предназначено для работы под управлением операционной системы Windows, которая пока не оптимизирована для функционирования в режиме жесткого реального времени. Кроме того, операторская станция обычно отличается нестабильным программным окружением (оператор может, например, запустить зараженную вирусом программу), что в сочетании со сложностью самой операционной системы может нарушить функционирование приложения, отвечающего за управление технологическим процессом вплоть до полного «зависания» компьютера с хорошо знакомым для многих визуальным эффектом «голубого экрана». Рассмотрим два возможных пути для перехода к системе, построенной по классической схеме «верхний уровень» + «слепые узлы».

В первом случае (рис. 5) в сети присутствуют три типа устройств: один-единственный ведущий контроллер (PLC-или IBM PC-совместимый контроллер), одна или несколько рабочих станций верхнего уровня, выполняющих роль операторских станций, серверов архивации или шлюзов для связи с локальной сетью предприятия, и необходимое количество распределенных по территории цеха или предприятия устройств ввода/вывода (контроллеры WAGO I/O фирмы WAGO, ET200 фирмы Siemens, а также широкий спектр датчиков и исполнительных устройств других фирм, совместимых с протоколом PROFIBUS-DP). Единственным ведущим в этой сети является сетевая карта, установленная в контроллере.

Контроллер «видит» через окно двухпортовой памяти мастер-карты каналы ввода/вывода удаленных УСО и область памяти slave-карты рабочей станции. Программа, выполняющаяся в контроллере, пишется на любом процедурном языке программирования общего назначения или на одном из языков стандарта IEC 61131. С помощью

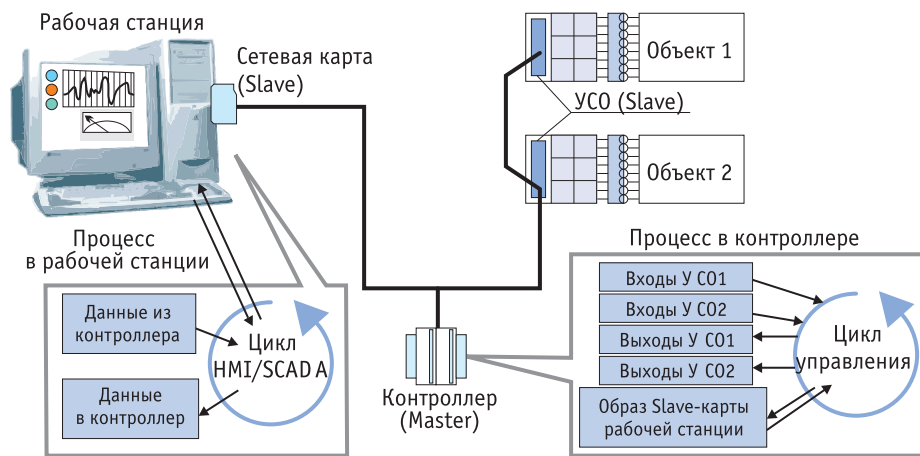


Рис. 5. Система с выделенным управляющим контроллером

специальных инструментальных средств (например, Ultra-logic) она работает в режиме реального времени и осуществляет основной цикл управления. В качестве аргументов при расчете управляющего вектора берутся значения с входных каналов УСО и дополнительные переменные (уставки или битовые комбинации с панели управления, «нарисованной» в SCADA-системе верхнего уровня), передаваемые с рабочей станции. В качестве результата расчетов в цикле управления получается управляющий вектор, направляемый на каналы вывода УСО, и дополнительный кадр выходных данных, посылаемый контроллером «наверх». Этими данными, записываемыми в область двухпортовой памяти соответствующей рабочей станции, в предельном случае может быть полный набор участвующих в процессе переменных, включая входные, выходные и расчетные. В этом случае SCADA получает полную информацию о процессе, но нельзя забывать, что за каждую точку ввода/вы-

вода нужно платить. Платить как в прямом смысле (стоимость всех современных SCADA-пакетов, например GENESIS32, напрямую зависит от числа контролируемых точек), так и в переносном (число передаваемых переменных увеличивает трафик сети и занимает ресурсы рабочей станции).

В связи с этим передавать в SCADA-систему желательно не все, а только то, что должно быть непосредственно видно оператору и сохраняться в архивах. Кроме SCADA-пакета, на рабочей станции должен быть установлен некий драйвер сети PROFIBUS-UDP. Для современных пакетов могут использоваться соответствующие OPC-серверы, поставляемые, например, фирмой Hilscher.

Таким образом, описанная система обеспечивает очень быстрый и фиксированный по времени цикл управления, гарантированную доставку сетевых пакетов и независимое функционирование SCADA-системы верхнего уровня.

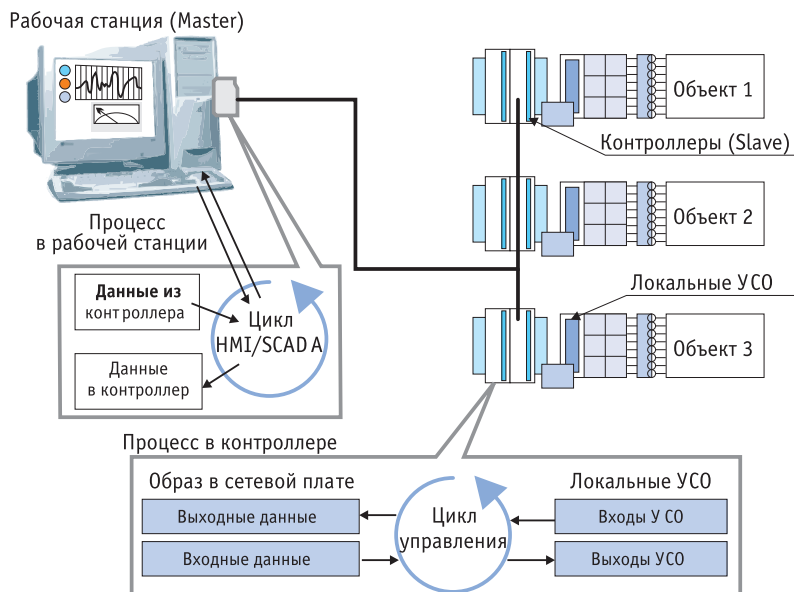


Рис. 6. Распределенное управление, локальные УСО

Во второй модели (рис. 6) рабочая станция является ведущей в сети, а контроллеры — ведомыми. В качестве контроллеров могут выступать, например, процессорные платы MicroPC с сетевыми адаптерами Hilscher. Все устройства ввода/вывода в данном случае являются локальными. Контроллеры, с одной стороны, выполняют ввод/вывод из локальных устройств, производят необходимые расчеты, осуществляют управление исполнительными устройствами, а с другой — публикуют все необходимые данные в сетевой плате PROFIBUS-DP (Slave). Ведущему (рабочей станции) остается собрать данные с контроллеров, передать им необходимые управляющие воздействия и организовать взаимодействие с оператором и архивом.

Данная архитектура также является полностью детерминированной и поддерживающей «распределенный интеллект», но в силу ограничения протокола PROFIBUS-DP не позволяет использовать в циклах управления удаленные переменные от других контроллеров без участия SCADA-системы. Это связано

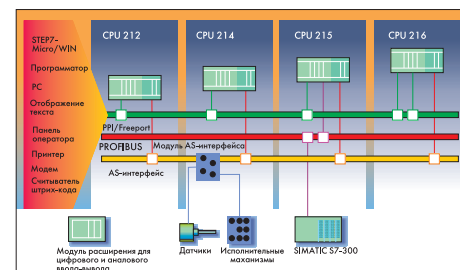


Рис. 7. Возможность работы контроллеров одновременно в режимах «Master AS-i» и «Slave PROFIBUS-DP»

с тем, что в сети PROFIBUS-DP может быть только один ведущий.

Большой выбор аппаратных средств, оконечных устройств и программного обеспечения делает решения на базе PROFIBUS одними из самых распространенных.

Необходимо упомянуть, что контроллеры и компьютеры, объединенные промышленной сетью, могут одновременно выполнять роль шлюзов в сети других уровней. Например, один из slave-контроллеров PROFIBUS (рис. 7) может одновременно быть мастером для сети более низкого уровня, связывающей элементарные датчики, УСО или исполнительные механизмы (сети AS-i, Seriplex, простые протоколы на базе RS-485 и т. п.). В то же время информация с уровня АСУ ТП должна поступать на уровень управления предприятием в целом (АСУП), где в подавляющем числе случаев применяется Ethernet.

ЗАКЛЮЧЕНИЕ

Хотя различные конфигурации технических средств описывались в данной статье на примере PROFIBUS-DP, многие соображения справедливы для распределенных систем управления, использующих другие разновидности промышленных сетей. Мы планируем описать их в следующей статье.