

ПРИЛОЖЕНИЯ

Приложение 1. Стандартный интерфейс общения программ – OLE

OLE – Object Linking and Embedding (связывание и внедрение объектов). Данная, разработанная и стандартизованная Microsoft технология позволяет в среде Windows обмениваться объектами (программами) между программой–поставщиком (сервером OLE) и программой–получателем (клиентом OLE). Она определяет три вида обмена:

– копирование объекта – одномоментное действие, при котором объект теряет связь с сервером и переходит к клиенту;

– внедрение объекта (Embedding) – действие, при котором объект переходит к клиенту, а последний запоминает сервер и, при необходимости, редактировать объект он обращается к серверу для проведения этого действия;

– связывание объекта (Linking) – действие, при котором объект не переходит к клиенту, а последний хранит о нем визуальное представление и его адрес в сервере. Если в сервере объект изменился, то и клиент будет его иметь в измененном виде.

Действия внедрения и связывания позволяют клиенту обрабатывать объект не у себя, а в сервере; при этом связывание экономит память у клиента, поскольку он хранит не сам объект, а только его адрес в сервере.

Приложение 2. Стандартный межпрограммный протокол – DDE/NetDDE и компонентная объектная технология COM/DCOM

Описание стандартного межпрограммного протокола –DDE/NetDDE.

DDE/NetDDE – Dynamic Data Exchange (разработанный и стандартизованный Microsoft аппарат динамического обмена данными между программами в среде Windows, локальный и сетевой). Программы, используя DDE, посылают данные другим программам и принимают данные от них. DDE создает "горячие связи", позволяющие одной программе информировать другие программы об изменении данных. Через протокол DDE программа может выполнить команду в другой программе.

При необходимости взаимодействия программ на разных узлах сети (разных компьютерах) используется сетевой протокол NetDDE, обладающий теми же свойствами, что и DDE.

Следует иметь в виду, что рассматриваемый протокол работает достаточно медленно и он не очень надежен в работе. Последнее время все более распространяется другой механизм взаимодействия программ по технологии COM/DCOM.

Описание компонентной объектной технологии COM/DCOM.

COM/DCOM – Component Object Model/Distributed Component Object Model (компонентная объектная модель, локальная и распределенная). Она разработана и стандартизована Microsoft для среды Windows, но постепенно переносится на другие распространенные операционные системы. По этой технологии программа строится из взаимодействующих между собою отдельных объектов-компонентов. Каждый компонент является самостоятельной, скомпилированной

программой, готовой к использованию. При наличии библиотеки различных компонентов, можно из них собирать требуемые программы. Стандарт COM/DCOM определяет технологию создания программ из отдельных объектов, разработанных по определенным правилам; другими словами, COM/DCOM устанавливает правила, которым должны следовать все компоненты, чтобы обеспечить их совместную работу. Распределенная компонентная архитектура DCOM поддерживает множество распространенных сетевых протоколов: TCP/IP, UDP, IPX/SPX, NetBIOS и др., поэтому программы, использующие технологию DCOM, могут работать в различных типах сетей.

Основные свойства компонентов:

- компонент является скомпилированной, готовой к реализации программой;
- компонент состоит из функциональной программы и интерфейса. Любая модификация функции, реализуемой компонентом, не затрагивает интерфейса, благодаря этому любое изменение компонента не нарушает его связи с другими компонентами;
- компонент может быть написан на любом языке программирования, это не сказывается на его связях с другими компонентами;
- компоненты могут находиться на одном компьютере или на разных узлах сети, это никак не сказывается на их взаимодействии.

Для организации межкомпонентного взаимодействия используют механизм локального вызова процедуры, если компоненты находятся на одной машине, и механизм удаленного вызова процедуры, если они находятся на разных машинах. При вызове одним компонентом другого он передает ему необходимые исходные данные для реализации заложенной в компоненте функции; если компоненты находятся на разных машинах, то эти данные преобразуются операционной системой в формат, учитывающий межмашинные различия. Для самих компонентов взаимодействие внутри одной машины или на разных узлах сети не отличаются, все связи и преобразования автоматически осуществляются стандартом COM/DCOM.

Приложение 3. Обмен программ с СУБД на базе стандартного драйвера ODBC

ODBC – Open DataBase Connectivity (открытое взаимодействие баз данных). Стандарт Microsoft – ODBC позволяет взаимодействовать приложениям (программам), работающим в среде Windows, посредством операторов языка SQL с различными СУБД, функционирующими под различными операционными системами. Фактически, ODBC это интерфейс, обеспечивающий взаимную совместимость серверных и клиентских компонентов доступа пользователя к данным.

Для реализации унифицированного доступа к различным СУБД введено понятие драйвера ODBC. Драйвер ODBC состоит из клиентской и серверной частей. Клиентская часть состоит из менеджера драйверов и ODBC-драйверов. Менеджер драйверов, получив запрос на функции ODBC по выполнению SQL-инструкций, загружает соответствующий определенной СУБД ODBC-драйвер. ODBC-драйвер обрабатывает функции ODBC, передает операторы SQL в СУБД и возвращает результат отправителю запроса (пользователю или прикладной программе). Серверная часть, находящаяся на стороне СУБД, – источник данных, в ней находятся детали сетевого интерфейса, расположение и имя СУБД и другие необходимые для общения с клиентской частью элементы.

Приложение 4. Стандартный язык запросов к реляционным СУБД – SQL

SQL – Structured Query Language (реляционный структурированный язык запросов). Это международный стандарт, первая версия которого была утверждена в 1989 г. В настоящее время он поддерживается подавляющим большинством СУБД, которые имеют для этого компилятор запросов языка SQL. В целом, язык SQL является универсальным средством общения пользователей и их прикладных программ с СУБД.

Язык SQL строится как логическое условие выборки определенных данных из одной или ряда таблиц (файлов) СУБД; он базируется на широком использовании различных предикатов и кванторов.

Язык SQL обеспечивает авторизацию доступа к СУБД: каждый пользователь имеет свои, доступные ему объекты базы данных и он, в частности, может с помощью SQL передать свои права на эти объекты другому пользователю.

Язык SQL включает средства динамической компиляции запросов. Допускается динамическая параметризация статически откомпилированных запросов, т. е. возможно построение эффективных диалоговых систем с типовыми наборами параметризуемых запросов.

Приложение 5. Взаимодействие на базе стандартной архитектуры ActiveX

На технологии Microsoft – COM/DCOM базируется архитектура ActiveX, которая позволяет в среде Windows одному программному компоненту (так называемому, управляющему элементу ActiveX или объекту ActiveX) взаимодействовать с другими программными компонентами.

Объекты ActiveX реализуются в двух основных формах: как встроенные в программу серверы и как серверы, реализуемые отдельно от программы. В первом случае программные компоненты могут строиться как ActiveX контейнеры (клиенты), в которые встраиваются серверы – управляющие элементы ActiveX. Во втором случае объекты ActiveX являются основой создания распределенных программ. При реализации объекта ActiveX как встраиваемого сервера достигается максимальная производительность, при реализации объекта ActiveX вне пространства программы уменьшается производительность, и увеличиваются накладные расходы.

Объекты ActiveX позволяют программе:

- получать запрашиваемую информацию с разных программ и узлов сети;
- загружать и запускать программы на разных узлах сети;
- выполнять заданные функции в сетевой среде;
- получать извещения о происшедших событиях на разных узлах сети.

Обычно объекты ActiveX пишутся на языках Visual Basic или C/C++, а используются в программах, написанных на разных языках.

Приложение 6. Intranet технология

Наряду с вышеуказанными типовыми интерфейсами, открытость и простота получения данных в информационных сетях начинает достигаться все более широким внедрением Intranet технологии. Возможность любым конечным пользователям общаться с другими пользователями, с программами и с базами данных средствами Internet'a существенно сказывается на скорости внедрения, простоте и легкости эксплуатации информационных систем.

Intranet – это корпоративный инструмент для обмена информацией на базе стандартных технологий Internet'a: WEB серверов, протоколов TCP/IP, языка гипертекста HTML. В сети Intranet каждый пользователь имеет у себя программу (Browser), которая понимает гипертекст, на котором она общается с WEB-сервером. Он, в свою очередь, через специальные интерфейсы общается с другими серверами сети: СУБД, хранилищами данных, серверами приложений и др., обеспечивая все потребности пользователей. Intranet реализует оперативную доставку информации с помощью электронной почты; позволяет:

- просто и удобно взаимодействовать пользователям на различных аппаратных средствах;
- использовать для общения с СУБД простейший язык – гипертекст;
- работать с прикладными программами на языке Java, исполняемыми на любой аппаратной платформе и под любой операционной системой;
- непосредственно со своих WEB-страниц иметь доступ в глобальную сеть Internet.

Приложение 7. Стандартный интерфейс взаимодействия программ в промышленных системах автоматизации – OPC

Стандарт OPC (OLE for Process Control – стандарт OLE для промышленного управления) определяет, в частности, механизм доступа к данным контроллера из любого программного приложения. Наличие во всех перспективных открытых SCADA программах стандартного интерфейса OPC позволяет непосредственно информационно соединять их с контроллерами разных фирм, для которых имеется интерфейс OPC без разработки специальных драйверов.

OPC базируется на общих технологиях OLE и DCOM и работает в среде Windows. Он разработан независимой международной организацией OPC Foundation.

В целом, стандарт OPC обеспечивает совместимость и взаимозаменяемость средств автоматизации от разных производителей. Через интерфейсы OPC одни программы могут читать и записывать данные в другие программы, передавать различные сообщения друг другу; все это происходит независимо от расположения программ на различных узлах сети и независимо от производителей аппаратуры и самих, находящихся в ней программ.

Взаимодействие по интерфейсу OPC основано на клиент-серверной схеме. OPC-клиент может работать только там, где установлена объектная модель DCOM; т.е. в операционных системах Windows NT, Windows 95/98/2000, отдельных версиях Unix, Windows CE. В подавляющем большинстве случаев, этими операционными системами оснащены рабочие станции операторов, и именно их программное обеспечение (SCADA-программы, различные прикладные пакеты программ) может выполнять роль клиента при обмене данными с любым устройством (контроллером, блоком ввода/вывода, интеллектуальным прибором), для которого есть OPC-сервер. При этом реализуются как прямые вызовы от клиента к серверу, так и обратные – от сервера к клиенту. Таким образом, любое устройство, для которого имеется OPC-сервер, может

без каких–либо специальных драйверов взаимодействовать с любой перспективной открытой SCADA-программой, работающей под операционной системой, с установленной моделью DCOM.

Следует отметить, что хотя OPC не обеспечивает работу в жестком реальном времени, имеющаяся частота передачи данных порядка 50 мс обычно удовлетворяет требованиям большинства конкретных промышленных объектов.

Стандарт OPC состоит из трех спецификаций:

- доступ к данным реального времени,
- обработка тревог и событий,
- доступ к историческим данным.

Доступ к данным реального времени имеет основное понятие – элемент данных. Каждый элемент данных (измеряемая величина) состоит из значения, которое может быть любой информацией скалярного типа; времени последнего обновления, которое проставляется с 100-наносекундной точностью; признака качества – кода, который определяет достоверность значения.

Метка времени либо формируется и передается контроллером, либо вносится компьютером, получающим данные.

Метка качества определяет качество данных и может принимать значения: хорошо, неопределенно, плохо. Если качество плохое, то получившее эти данные приложение имеет, обычно, доступ к полям диагностики, где метка качества расшифровывается. Она может конкретизироваться одним из ряда причин:

- отсутствие связи,
- сбой устройства,
- ошибка конфигурации,
- отказ датчика,
- данные заблокированы,
- причины неизвестны.

По требованию клиента OPC–сервер создает *группу* данных, которую он обновляет и передает клиенту с заданной ему клиентом частотой. Клиент может создавать для себя на сервере несколько групп данных, различающихся требуемой частотой обновления.

Обработка тревог и событий производится OPC-сервером тревог; он формирует определенные логические параметры (события), посылает их клиентам, а последние подтверждают получение этих сообщений.

Доступ к историческим данным производится через OPC-сервер исторических данных, который полученные в реальном времени значения величин архивирует и предоставляет их по запросам клиентов.

Обмен данными между клиентом и сервером может проходить в синхронном режиме (клиент инициирует периодический обмен) и асинхронном режиме (сервер оповещает клиента об

изменившихся значениях величин, интересующих клиента, что обеспечивает минимальное время обнаружения разных событий).

В перспективных открытых SCADA-программах интерфейс OPC может быть включен либо как один из интерфейсов взаимодействия с внешними программами; либо являться основой структуры SCADA-программы. В последнем случае все компоненты SCADA-программы взаимодействуют между собою через OPC, являясь, в зависимости от ситуации клиентами или серверами, или теми и другими одновременно.

Инструментальные средства для разработки OPC-компонентов могут либо поставляться разработчиками SCADA программ, либо независимыми производителями программных средств. Использование специализированных инструментальных средств создания OPC-серверов и OPC-клиентов значительно упрощает разработку OPC-компонентов, поскольку предлагает готовую реализацию OPC-интерфейса.

Приложение 8. Стандартная сеть с HART-протоколом

Широко распространенный и поддерживаемый разными фирмами, выпускающими, как контроллеры, так и интеллектуальные приборы, протокол HART (Highway Addressable Remote Transducer) был разработан фирмой Rosemount в середине 80-х годов и стандартизирован в Америке. Технология использования протокола, координирование и поддержка его применения обеспечиваются независимым Фондом HART коммуникаций. Множество производителей контроллеров и интеллектуальных приборов в разных странах поддерживают этот протокол.

Этот протокол основан на аналоговой 4–20 мА-технологии. Он нацелен на связь контроллера с интеллектуальными приборами и имеет два варианта их связи.

При первом варианте реализуется связь каждого прибора с контроллером по отдельной паре проводов, по которой могут проходить как аналоговый (например, 4–20 мА), так и цифровые сигналы. Последние содержат дополнительную информацию о работе прибора: диапазон и единицы измерения, дату калибровки, самодиагностику и т.д. Питание прибора осуществляется от блока питания в контроллере по этой же паре проводов.

При втором варианте связи реализуется соединение ряда приборов с контроллером по одной паре проводов, в этом случае по паре проводов могут проходить только цифровые сигналы. Следовательно, в каждом приборе аналоговые выходы должны быть преобразованы в цифровой код.

Основные характеристики сети:

- метод доступа – ведущий/ведомый. В сети может быть до двух ведущих узлов; ими бывают контроллеры и ручные коммуникаторы. Один ведущий узел реализует циклы передачи и запросы к ведомым узлам; другой (если он есть) – используется для связи сети с другой системой;

- физическая среда передачи: последовательный порт RS–232 C и витая пара;

- топология сети: звезда, при соединении к контроллеру через сеть датчиков с аналоговыми и цифровыми сигналами, и шина – при только цифровых сигналах, проходящих через сеть;

- режимы работы: асинхронный, когда ведущий узел посылает запрос, а ведомый – ответ (цикл укладывается в 500 мс); синхронный, когда ведомые узлы непрерывно передают свои данные ведущему узлу; (время обновления данных в контроллере – 250–300 мс);

– при шинной архитектуре HART протокол работает в режиме моноканала (т. е. более двух устройств соединяются через один канал передачи) и к сети может быть подключено до 15 ведомых узлов (обычно подключают до 8 приборов);

– длина линии связи до 1500 м;

– скорость передачи данных – 1200 бит/с;

– протокол HART реализует по модели OSI следующие уровни: первый (физический), второй (канальный), седьмой (прикладной);

– команды от ведущих узлов могут быть трех видов: универсальные на все ведомые узлы, типовые на многие ведомые узлы и специфические на конкретные ведомые узлы;

– каждое сообщение от прибора содержит информацию двух типов: текущие данные и статус прибора. Последняя информация определяет оперативное состояние прибора: нормальный последовательный статус, когда данные от него могут быть использованы для вычислений и управления; нормальный непоследовательный статус, когда данные от него корректны, но с прибором связана какая-то тревога: неопределенный статус, когда данные не полностью корректны, но все же могут быть использованы; плохой статус, когда данные не могут быть использованы;

– за каждую посылку информации любое устройство может передать другому устройству 4 технологических переменных, а само устройство может иметь до 256 переменных, описывающих его состояние.

Приложение 9. Стандартные сети Foundation Fieldbus

Сети образованы двумя ведущими американскими ассоциациями ISP и WorldFIP, которые объединились с другими фирмами в 1994 году в ассоциацию Fieldbus Foundation. Эта ассоциация опубликовала и поддерживает стандарт на сети и производит сертификацию устройств на соответствие стандарту. Сейчас в ассоциацию Fieldbus Foundation входят более 100 крупнейших компаний, которые представляют порядка 90% производителей средств и услуг в области автоматизации.

Нацеленность сетей – нижний уровень распределенной системы автоматизации (связь контроллеров с выносными блоками ввода/вывода и с интеллектуальными приборами) при учете специфики работы приборов во взрывоопасной среде.

На нижнем уровне управления – приборном уровне (датчики и исполнительные механизмы) в последние годы начали происходить крупные изменения. Эти изменения с одной стороны обусловлены общими тенденциями стандартизации различных классов средств автоматизации, а с другой стороны – увеличением мощности и параллельным удешевлением микропроцессоров, встраиваемых в отдельные приборы.

Для более полного и рационального использования все возрастающей мощности микропроцессоров, встраиваемых в интеллектуальные приборы, разработана идеология Fieldbus Foundation, которая ставит своей целью перенос типовых алгоритмов переработки измерительной информации (фильтрации, масштабирования, линеаризации и т. п.), регулирования (стабилизации, слежения, каскадного управления и т. п.), логического управления (пуска, останова, блокировки группы механизмов и т. п.) на самый нижний уровень управления: уровень интеллектуальных датчиков и исполнительных механизмов. Для реализации этой идеологии разработан новый по возможностям и параметрам стандарт на полевую сеть.

Основная особенность стандарта, отличающая построенные по нему сети от всех других распространенных полевых сетей, разработка условий работы и обмена информацией между приборами в сети при учете, что каждый прибор в сети, кроме обычных функций аналого-цифрового или цифро-аналогового преобразований может производить типовые функции контроля и управления.

Основной вариант сети – Foundation Fieldbus H1 (FF H1); он реализует безопасную работу приборов во взрывоопасной среде. Кроме него существует вариант Foundation Fieldbus H2 (FF H2), в котором специфика работы приборов во взрывоопасной среде не учитывается.

Характеристики сети FF H1

Ниже описываются основные свойства и характеристики сети, обозначаемой как FF H1.

Топология сети – шина или дерево.

Физическая среда – витая пара.

Длина линии передачи – 1,9 км.

Скорость передачи данных – 31,25 кбит/с.

Число подключаемых к сети устройств – до 32.

Сеть используется также для передачи питания от контроллера к приборам, подключенным к сети.

Протокол сети использует три уровня по модели OSI: первый (физический уровень), второй (канальный уровень) и седьмой (прикладной уровень). Кроме того, стандарт Foundation Fieldbus особое внимание обращает на пользовательский уровень (он как бы над прикладным уровнем), на котором фиксируется ряд важных функций и правил.

Метод доступа к сети – маркер. Активный планировщик связей (LAS – Link Active Scheduler), работающий как арбитр сети, поддерживает плотный временной график периодических сообщений. Он же обеспечивает быстрый доступ к сети для высокоприоритетных асинхронных событий (сеть работает с приоритетными сообщениями). Управление сетью может быть распределено между несколькими активными планировщиками связей; они могут резервировать друг друга.

Периодический цикл передачи информации с учетом отработки в устройствах контуров управления составляет 50 мс.

Введенный в стандарт FF H1 пользовательский уровень (отсутствующий в других стандартах и в модели OSI) имеет несколько важнейших черт.

Он определяет связи, с помощью которых пользователь может взаимодействовать с приборами через серию так называемых блоков, причем удобнее и скорее чем с запросами по отдельным точкам. Пользовательский уровень может использовать три типа блоков:

- блоки приборов – описывают такие характеристики приборов как имя, производитель, номер серии и т. п.;
- блоки функций – определяют работу приборов по вводу сигналов, контролю и управлению, выводу сигналов. Всего стандарт определяет 10 функциональных базовых блоков: аналоговый вход, аналоговый выход, смещение, переключатель управления,

дискретный вход, дискретный выход, ручной загрузчик, регулятор ПД, регулятор ПИД, регулятор отношения. Любые другие функциональные блоки могут быть определены конкретными производителями приборов и согласованы с ассоциацией Fieldbus Foundation;

– блоки преобразователей – расщепляют отдельные блоки функций на чтение/запись локальных входных/выходных данных.

Важно подчеркнуть, что функциональные блоки могут при их соответствующем соединении друг с другом реализовывать простейшие цепи управления, как в отдельном приборе, так и распределение через сеть в нескольких приборах.

Вторая важная черта пользовательского уровня стандарта FF H1 – описатели приборов (DD – Device Descriptions). DD определяет стандартное описание функций, которые можно реализовывать в приборе. Используя DD, оператор может взаимодействовать с прибором: конфигурировать калибровку, менять параметры, диагностировать работу и т. п. Механизм DD дает оператору возможность полностью определять, конкретизировать и модифицировать свойства прибора.

Для работы с функциональными блоками в приборе используется специальный язык описания устройств (DDL – Device Description Language), который специфицирован в ассоциации Fieldbus Foundation. Он описывает функциональные блоки, используется для доступа к информации в приборе и для определения дополнительных характеристик, которые можно добавить к функциональному блоку. Используя описатели приборов DD, язык DDL позволяет оператору составлять алгоритм работы прибора и полностью контролировать его работу.

Аналогично HART протоколу в полевой сети FF H1 используется понятие "статус", которое каждый цикл передается каждым прибором по сети вместе с его данными. Статус определяет оперативное состояние прибора: нормальный последовательный статус, когда данные от него могут быть использованы для вычислений и управления; нормальный непоследовательный статус, когда данные от него корректны, но с прибором связана какая-то тревога: неопределенный статус, когда данные не полностью корректны, но все же могут быть использованы; плохой статус, когда данные не могут быть использованы. Каждое значение статуса имеет 16 различных подстатусов, которые конкретизируют и диагностируют имеющийся статус, т. е. определяют уровень самодиагностики приборов FF H1. С помощью DD можно добавить дополнительные диагностические функции.

Характеристики сети FF H2

Вариант сети FF H2 отличается от рассмотренного варианта FF H1 следующими характеристиками:

- топология сети – шина;
- физическая среда – витая пара или кабель;
- длина линии передачи – 0,75 км;
- скорость передачи данных – 1,0 или 2,5 Мбита/с;
- питание приборов через шину не производится;

– сеть не предназначена для работы во взрывоопасной среде. Остальные характеристики аналогичны сети FF H1.

Приложение 10. Стандартные сети PROFIBUS

Данные сети с начала 90-х годов получают все более широкое распространение среди европейских производителей средств автоматизации. Специальная организация пользователей сетей PROFIBUS PUO (Profibus User Organization) проводит сертификацию устройств на соответствие утвержденному стандарту PROFIBUS. Более 1600 заводов, исследовательских институтов, заказчиков средств, автоматизации входят в эту организацию.

Использование сети на разных уровнях и в разных условиях обеспечивается тем, что сеть поддерживает три разных вида протоколов:

- протокол Profibus DP для децентрализованного соединения и быстрой коммуникации блоков ввода/вывода и интеллектуальных приборов;
- протокол Profibus FMS для решения сложных коммуникационных задач на контроллерном уровне управления;
- протокол Profibus PA для повышенных требований к безопасности.

Все три вида протокола используют общий канальный уровень (второй уровень по модели OSI), а протоколы Profibus DP и Profibus FMS имеют одинаковые первый (физический) и второй (канальный) уровни протокола.

Характеристики сети PROFIBUS–DP

Сеть предназначена для связи выносных блоков ввода/вывода и интеллектуальных приборов с контроллерами.

Ниже перечислены основные характеристики сети.

Физически передача данных в сети осуществляется через порт RS–485 и экранированную витую пару длиной до 9,6 км, либо через оптоволоконный кабель длиной до 90 км.

Сеть обеспечивает любые виды соединений: шина, дерево, звезда, кольцо.

Метод доступа к сети: ведущий/ведомый. В сети возможно наличие нескольких ведущих устройств; при этом либо каждое ведущее устройство обслуживает свои ведомые, либо одно из них организует работу ведомых, а другие диагностируют, конфигурируют и производят другие фоновые операции. Записывать данные в ведомое устройство может только одно ведущее устройство. Сами ведущие узлы общаются между собой с помощью маркера.

Общее число устройств на сети до 126, из которых 32 узла могут быть ведущими.

Скорость передачи данных на витой паре варьируется от 9,6 Кбит/с до 1,5 Мбит/с. При длине витой пары 200 м скорость максимальна 1,5 Мбит/с.

Длина сети на витой паре до 1,2 км или до 4,8 км с повторителями. Длина сети на оптоволокне – до 23 км.

Основа работы протокола – циклический опрос ведомых устройств; кроме того, существуют ациклические функции диагностики, конфигурирования диапазонов измерения и т. п.

Передаваемые устройствами диагностические сообщения имеют три уровня иерархии: диагностика всего устройства (например, упало напряжение питания), диагностика модуля

устройства (например, отказал 8-ми канальный цифровой модуль выходных сигналов), диагностика канала (например, в канале А модуля Б не проходит сигнал).

В сети реализована коррекция ошибок: в любой посылке данных 3 ошибочных бита будут обнаружены, а 1 ошибочный бит может быть восстановлен.

В каждом сеансе связи ведомое устройство может передать до 246 байт информации, обычно передается 32 байта.

При скорости 1,5 Мбит/с передача 512 бит данных, распределенных между 32 устройствами, занимает 6 мс; та же передача при скорости 0,5 Мбит/с занимает 18 мс.

Характеристики сети PROFIBUS–FMS

Этот протокол предназначен для работы на цеховом (контроллерном) уровне для соединения контроллеров между собою и с рабочими станциями операторов. На этом уровне часто обмен информацией осуществляется по запросу прикладного процесса и не является циклическим, поэтому время реакции здесь может быть менее существенно, чем функциональные возможности сети. Протокол оперирует с передачей массивов информации и воспринимает понятия переменная, массив, запись, область памяти, событие, проводит логическую адресацию.

Основные характеристики соответствуют протоколу Profibus DP, скорость передачи данных на витой паре до 500 Кбит/с.

Характеристики сети PROFIBUS–PA

Специализированный протокол для передачи данных от выносных блоков ввода/вывода и интеллектуальных приборов к контроллерам во взрывоопасных средах. По протоколу канального уровня Profibus PA полностью идентичен Profibus DP, но он имеет иную физическую реализацию: безопасное низковольтное исполнение. Коммутация устройств реализуется на одной витой паре, которая одновременно используется для информационного обмена и для питания устройств. Основные области применения этого протокола – предприятия химии, нефтехимии, нефтепереработки.

Скорость передачи данных на витой паре до 31 Кбит/с.

Приложение 11. Характеристики промышленных сетей

Стандартные физические порты, используемые промышленными сетями: RS–232, RS–422, RS–485. Подавляющее большинство сетей использует наиболее мощный по числу узлов на сети и наиболее скоростной по передаче данных порт RS–485. Ниже дано описание стандарта данных портов.

Описание стандарта на порты последовательной передачи данных – EIA

Подавляющее большинство современных компьютеров, контроллеров и интеллектуальных приборов имеют в своем составе порты последовательного ввода/вывода данных RS–232, RS–422 и RS–485, которые применяются для соединения этих средств друг с другом. Стандарт EIA устанавливает требования к электрическим параметрам передатчиков и приемников двоичных цифровых сигналов, которые объединены в систему связи подключением к общему кабелю. Он не касается программных протоколов. В таблице 8 приведены главные черты этих трех последовательных портов, как они определены в документах стандарта.

Таблица 8.

| | <i>RS-232</i> | <i>RS-422</i> | <i>RS-485</i> |
|---------------------------------------|--------------------|------------------|------------------|
| Тип передающей линии | Несбалансированный | Дифференциальный | Дифференциальный |
| Максимальное число передатчиков | 1 | 1 | 32 |
| Максимальное число приемников | 1 | 10 | 32 |
| Максимальная длина линии связи, м | 15,2 | 1200 | 1200 |
| Максимальная скорость передачи данных | 20 Кбит/с | 10 Мбит/с | 10 Мбит/с |

RS-232 широко используется в промышленных средствах автоматизации. Он обеспечивает соединение «точка к точке» между последовательным портом контроллера и приборами. Часто, за счет усовершенствования передатчика и кабеля, достигается большая длина линии и скорость, чем зафиксировано в стандарте.

RS-422 и RS-485 имеют, в отличие от RS-232, дифференциальный электрический сигнал передачи, который использует двухпроводную связь так, что по каждому проводу проходят передающие и обратные приемные сигналы. Это дает возможность резко поднять устойчивость передачи к помехам и увеличить длину линии связи по сравнению с интерфейсом RS-232. Так, устройства, соответствующие стандарту RS-485, работоспособны при воздействии на них таких не скомпенсированных разностей потенциалов земли передатчиков и приемников, напряжений помех, напряжений смещения выходов передатчиков, которые в совокупности лежат в диапазоне $(-7+7)$ В. Если значение разности потенциалов между землями выходит за пределы допустимого диапазона, то следует применять устройства с гальванической изоляцией.

RS-485 – самый совершенный из рассматриваемых интерфейсов (стандарт RS-485 выпущен последним в 1983 году) и имеет наибольшее число подключаемых к линии связи приборов. Он нашел наиболее широкое применение в промышленных средствах автоматизации.

Полная система связи интерфейса RS-485 включает в себя ряд передатчиков, ряд приемников (в одном узле системы возможно также сочетание передатчика и приемника), соединенных симметричным кабелем. Устанавливаемые стандартами электрические значения параметров для интерфейсов RS-422 и RS-485 близки; это обеспечивает возможность разработки передатчиков и приемников, соответствующих требованиям обоих стандартов. Электрические параметры указанных устройств в системе с интерфейсом RS-485 выбраны так, чтобы была возможность функционирования передатчика на эквивалентную нагрузку, соответствующую 32-м абстрактным устройствам, включенным в систему. Реальные устройства отличаются по эквивалентной нагрузке от абстрактных, поэтому конкретное максимальное число устройств, которое можно подключить к системе связи определяется статическими характеристиками нагрузки всех устройств системы.

Устройства, электрические параметры которых приведены в стандарте, могут быть применены для обмена данными в RS-485 при скорости передачи до 10 Мбит/с. Действительная скорость передачи данных определяется длиной и параметрами соединительного кабеля, степенью симметрии и качеством согласования линии связи. Тип кабеля определяется исходя из следующих вариантов требований:

- по имеющимся характеристикам устройств, требуемой протяженности линии связи, минимально допустимому уровню сигнала на входе наиболее удаленного приемника, максимально допустимому уровню искажений сигнала, выбирается кабель такого типа, который обеспечит максимально возможное значение скорости передачи данных;

- по имеющимся характеристикам устройств, требуемому значению скорости передачи данных, минимально допустимому уровню сигнала на входе наиболее удаленного приемника, максимально допустимому уровню искажений сигнала, выбирается кабель такого типа, который обеспечит максимально возможную протяженность линий связи.

Используемая топология промышленных сетей: звезда, кольцо, шина. Топология в виде звезды имеет ряд недостатков: доступ и управление реализуются через центральный узел, сбой центрального узла приводит к сбою системы, расширяемость системы ограничена числом портов в центральном узле. Кольцевая и шинная топологии лишены этих недостатков: доступ и управление децентрализованы, ошибка или отказ любого узла не приводит к сбою системы, расширение системы возможно; ввиду этого их использование является преобладающим.

Физическая среда передачи данных зависит от свойств конкретного производства и характеристик проектируемой системы; ею может быть витая пара, коаксиальный кабель, оптоволокно, радиоканал. В таблице 9 даны примерные характеристики этих физических сред.

Таблица 9.

| Характеристики | Витая пара | Кабель | Оптоволокно | Радиоканал |
|------------------------------------|---------------|----------|-------------|------------|
| Диапазон длины, км | 0,01–1,0 | 0,1–4,0 | 1,0–10,0 | 1,0–10,0 |
| Диапазон скорости, Мбит/с | 0,01–2,– 0 | 0,3–10,0 | 1,0–100,0 | 0,01–0,03 |
| Диапазон стоимости одного узла, \$ | 10–30 | 30–50 | 75–200 | 50–100 |
| Затраты на установку сети | Низкие | Средние | Высокие | – |

На предприятиях малого и среднего масштаба наибольшее распространение получают сети, работающие на витых парах.

Из зафиксированных в OSI-модели семи уровней сетевого протокола (в OSI-модели производится структурирование протокола на уровни, каждый из которых выполняет определенную функцию) в промышленных сетях обычно используется первый (физический уровень), второй (канальный уровень) и седьмой (прикладной уровень).

Применяемые методы доступа к промышленной сети, получившие наибольшее распространение:

- ведущий/ведомый (Master/Slave),
- передача маркера (the Token Passing Method),
- случайный метод доступа (Carrier Sense Multiple Access with Collision Detection CSMA/CD).

В методе «ведущий/ведомый» право инициировать циклы имеет только ведущий узел. Он запрашивает поочередно все ведомые узлы и передает их готовые к отправке данные. Ведущий узел может быть фиксированным или плавающим. Этот метод доступа является наиболее простым и распространенным в промышленных сетях.

В методе «передача маркера» право на доступ к шине (маркер) передается в цикле «от узла к узлу». Распределение шинных ресурсов между всеми узлами производится в соответствии с их запросами. Этот метод важен для применений, где надо гарантировать время передачи данных.

В случайном методе доступа каждый узел прослушивает шину и если она свободна, то он занимает ее. Если несколько узлов одновременно хотят занять шину (конфликтная ситуация), то они снимают свои заявки и включают случайные генераторы, которые задают случайный интервал ожидания до следующего момента запроса шины. Этот метод (наиболее широко распространенный в информационных сетях) применяется в промышленных сетях при возможности работы системы не в жестком реальном времени и при низкой загрузке шины порядка 10–30 % ее производительности (низкой вероятности конфликтных ситуаций).

Распространенные промышленные сети

Достаточно широкое распространение в ПТК различных фирм получили промышленные сети Modbus, Bitbus, Canbus, основные особенности которых приведены ниже.

Сеть Modbus имеет следующие основные характеристики: физические порты – RS–232C, RS–422, RS–485, токовая петля 20 ма; среда передачи – не специфицирована; метод доступа – ведущий/ведомый; число узлов – 1 ведущий и 247 ведомых; скорость передачи данных – 19,2 Кбит/с; длина сети до 1,2 км при RS–422, RS–485.

Сеть Bitbus имеет следующие основные характеристики: физический порт – RS–485; среда передачи – витая пара, оптоволокно, радиоканал; метод доступа – ведущий/ведомый; два режима передачи данных: режим с самосинхронизацией имеет до 250 узлов (до 28 узлов на сегмент сети), скорость передачи данных – 375 Кбит/с при длине сети до 0,3 км и – 62,5 Кбит/с при длине сети до 1,2 км; синхронный режим имеет до 28 узлов, скорость передачи данных от 500 Кбит/с до 2400 Кбит/с при длине сети 30м.

Сеть Canbus имеет следующие основные характеристики: среда передачи – витая пара; метод доступа – случайный (каждый узел имеет приоритеты сообщений и при возникновении коллизии выбирается сообщение с большим приоритетом); число узлов до 30; скорость передачи данных – 20 Кбит/с при длине сети до 1,0 км и – 1,0 Мбит/с при длине сети до 40 м.