

Тема 1.3. Распределенность компьютерных систем автоматизации

Распределенные системы компьютерной автоматизации и управления

В первой лекции упоминалось, что распределенные системы для целей автоматизации применять оказалось выгоднее, чем централизованные (сосредоточенные), поскольку централизованные системы того времени на базе ЭВМ (мейнфреймы) применялись лишь для очень сложных процессов в силу своей высокой стоимости (надежность современных мейнфреймов уже весьма высока). В настоящий момент централизованные системы применяются либо в очень простых процессах (это вызвано низкой стоимостью современных средств автоматизации в противовес мейнфреймам, но достаточностью ресурсов), элементы которого территориально сосредоточены для решения задач локального контроля и управления либо в специализированных процессах централизованной обработки информации, что, однако, лежит за пределами задач промышленной автоматизации. Решения простых задач локального контроля и управления уже недостаточно для комплексной автоматизации, да и процессы многих отраслей территориально распределены. Все это требует применения распределенных системы управления. Кроме того, **сосредоточенная система** является частью или частным случаем распределенной, поэтому появление распределенных систем является следствием естественного развития от частного к общему.

Таким образом, с ростом количества датчиков, увеличением площади территории, на которой расположена автоматизированная система и усложнением алгоритмов управления (требуется функциональное разделение) становится более эффективным применение **распределенных систем управления (PCY)**, т.е. проведение **топологической и функциональной децентрализации**. PCY применяют чаще всего в непрерывном и гибридном производстве (хотя, строго говоря, сфера применения PCY только этим не ограничена).

Распределенную систему управления (PCY, DCS — Distributed Control System) можно определить как *систему, состоящую из множества устройств, разнесенных в пространстве, каждое из которых не зависит от остальных, но взаимодействует с ними для выполнения общей задачи*. Под «множеством устройств», разнесенных в пространстве, понимается любое микропроцессорное устройство: контроллер, компьютер, модуль ввода-вывода и т.д. «Разнесение в пространстве» также не имеет границ, т.е. элементы системы могут находиться даже на разных континентах земного шара, а связь между ними может выполняться через Интернет.

При **территориальном (топологическом) распределении** контроллеров, модулей ввода-вывода и других устройств автоматизации структура распределенной системы и структура алгоритма ее работы становятся подобны структуре самого объекта автоматизации (рис. 1). Для обеспечения оптимальной топологической децентрализации число и места установки узлов системы выбираются так, чтобы минимизировать суммарную длину линий связи, достичь заданных показателей по надежности, скорости обмена и т. д. Основными топологиями построения являются шина, режé – кольцо и звезда (см. лекцию о промышленных сетях).

Целью **функциональной децентрализации** системы контроля и управления является снижение сложности системы путем разделения функций системы на более мелкие, т.е. разделение сложного и большого процесса на меньшие части – подпроцессы, при котором каждый контроллер работает со своей группой устройств ввода-вывода и обслуживает определенную часть объекта управления. При этом расчленение функций осуществляется так, чтобы обеспечить необходимую надежность и экономичность системы. В частности, технологическое оборудование, как правило, выпускается с уже встроенными ПЛК.

В ряде случаев технологический процесс легко разбивается на несколько слабо взаимосвязанных подпроцессов (функции сбора, обработки данных, управления и вычисления оказываются распределенными среди множества контроллеров), каждый из которых может быть реализован на отдельном контроллере или микро-ЭВМ, благодаря чему значительно снижается сложность вычислительной системы. Однако, другие технологические процессы более сложны, и их разбиение порождает сильно взаимосвязанные подпроцессы. При этом для упрощения решения задач в целом следует искать способы разбиения на подпроцессы, при которых будут обеспечены четко определенное и минимальное взаимодействие (обмен данными).

Распределение функций должно осуществляться в соответствии со следующими принципами:

- по технологическому тракту, когда конкретные функции относятся к конкретному мелкому агрегату или технологическому участку (например, станку или транспортному конвейеру);
- по режимам работы оборудования или фазам технологического процесса, когда конкретные функции относятся к более или менее самостоятельному режиму, являющемуся частью общего режима (например, аварийный режим или режим пуска и останова);
- по продолжительности интервала времени, необходимого для выполнения конкретной функции.

Максимальные преимущества распределенной системы достигаются, когда контроллеры работают автономно, а обмен информацией между ними сведен до минимума (хотя достаточный обмен информацией необходим для совместной реализации функций). Тенденция децентрализации управления и приближения контроллеров к объектам управления является общей для всех систем автоматизации. Распределенная система смягчает также требования к операционным системам (ОС) реального времени, поскольку задачи распределены между параллельно работающими контроллерами, каждый из которых может работать под управление своей ОС.

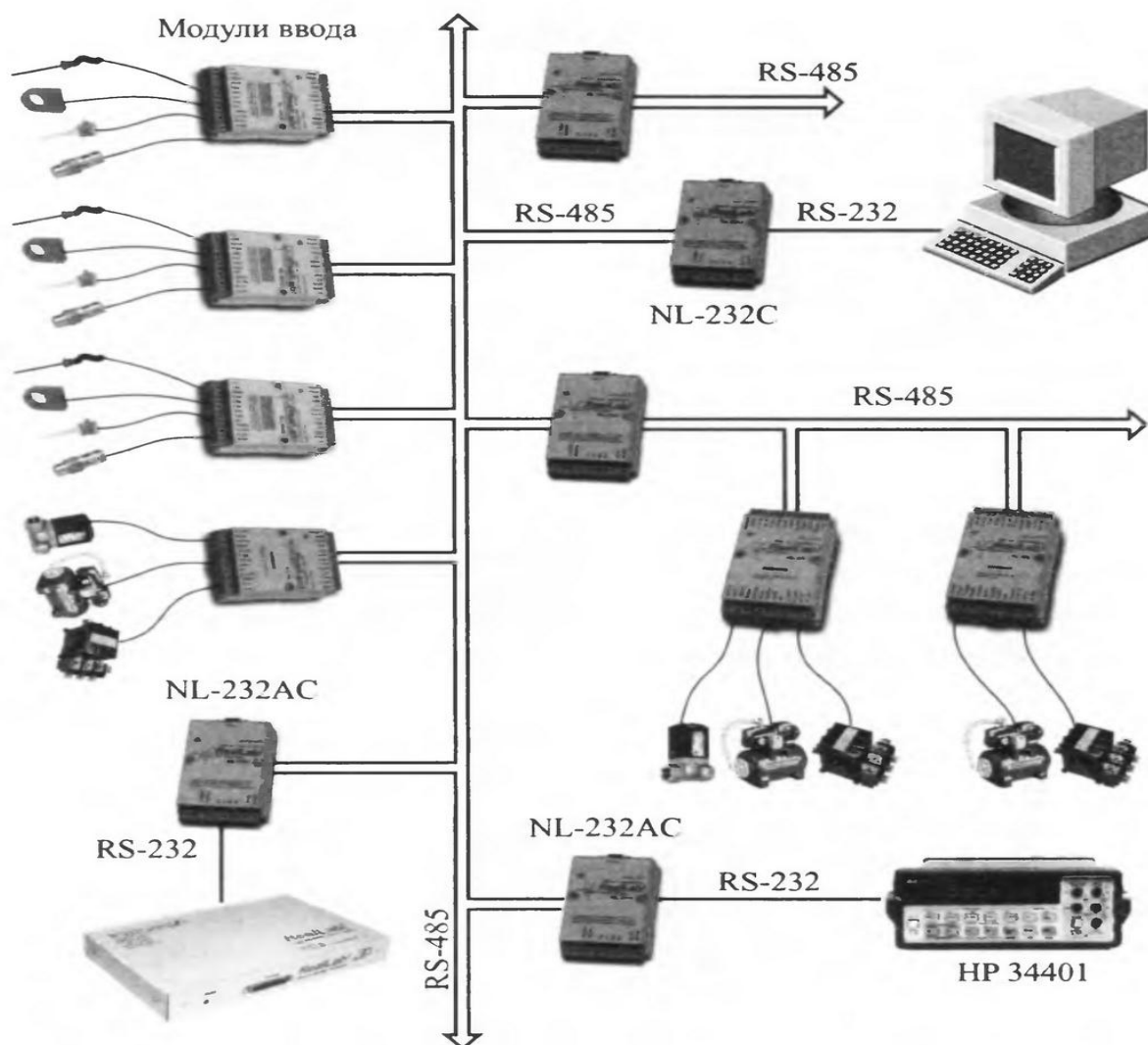


Рис. 1 – РСУ на базе устройств НИЛ АП

Требование автономности работы (т.е. автономного выполнения своих функций) и минимума обмена информацией не являются тривиальными, поскольку с ростом размера системы и увеличения количества ее элементов растет интенсивность потоков информации, а значит и задержка в передаче данных, что весьма опасно, например, при осуществлении регулирования. Дело в том, что большинство РСУ строятся по топологии «общая шина» (топологии кольцо и звезда используются значительно реже), что в отличие от топологии «точка-точка» подразумевает **необходимость адресации устройств и необходимость ожидания в очереди**. Для того чтобы получить данные из модуля или контроллера, компьютер (или контроллер) посылает в шину его адрес и команду запроса данных. Микропроцессор, входящий в состав каждого модуля или контроллера, сверяет адрес на шине с его собственным адресом, записанным в ПЗУ, и, если адреса совпадают, исполняет следующую за адресом команду. Команда позволяет считать данные, поступающие на вход устройства, или установить необходимые данные на его выходе.

Добавление адреса в коммуникационный пакет снижает скорость обмена при коротких сообщениях, а обмен по общей шине приводит к тому, что каждое устройство для передачи сообщения должно ждать, когда шина станет свободной. Это замедляет скорость обмена между устройствами по сравнению с топологией «точка-точка». Задержка в сетях с большим количеством устройств становится существенным ограничением на применение топологии с общей шиной в некоторых задачах, в частности в случае регулирования, когда задержка в сети ограничивает частоту работы контура регулирования. Конечно, регулирование – не единственная подобная задача, поэтому для

таких случаев используют локальные подсети (объектная декомпозиция) и локальные технологические контроллеры (функциональная декомпозиция), выполняющие функции резервирования, архивирования и многие другие.

Конкретно для реализации локального регулирования и простого управления часто используются так называемые **измерители-регуляторы** или просто регуляторы, т.е. совокупность модуля ввода, контроллера и модуля вывода выполненные в одном блоке и предназначенные для реализации позиционных и ПИД-закона регулирования определенной физической величины (терморегуляторы, регуляторы давления, расхода и т.д.). Для осуществления регулирования в них посылают значение уставки и параметры ПИД-регулятора (пропорциональный, дифференциальный и интегральный коэффициенты), а также другие настройки, затем команду запуска процесса регулирования. Информация в модулях распределенной системы предварительно обрабатывается (фильтруется, линеаризуется, корректируется и т.д.), а наличие ПИД-регулятора в модулях позволяет осуществить локальное регулирование (например, поддержание стабильной температуры в камере тепла и холода), разгрузив общую шину для выполнения других задач.

Поскольку устройство является гибридным, то часто продаются как интеллектуальные модули ввода-вывода или модули ввода-вывода с функцией регулирования. Примерами устройств могут быть модули NL-8TI, NL-16AI фирмы НИЛ АП, регуляторы температуры ТРМ210, ТРМ 101 фирмы ОВЕН.

Из вышесказанного вытекает, что часть устройств распределенной КСА должны, несмотря на необходимость адресации устройств и ожидания в очереди, работать в режиме **жесткого реального времени**, т.е. *гарантированно обмениваться информацией, а также вырабатывать в соответствии с алгоритмом и реализовывать управляющее воздействие в заданный промежуток времени*. Работа в реальном времени реализуется как аппаратно, так и программно и является необходимым требованием ко всем АСУ (части устройств), а не только распределенным КСА. Подробнее этот вопрос рассматривается в последующих лекциях.

С ростом количества датчиков в системе, увеличивается число и суммарная длина проводов, соединяющих датчики с устройством ввода. Это приводит не только к росту стоимости кабельного оборудования, но и к проблемам, связанным с электромагнитными наводками, особенно если датчики распределены по большой площади (например, в промышленной теплице датчики распределены по площади около 6 Га, а в элеваторе число датчиков достигает 3...5 тыс. шт.). В распределенной системе модули ввода-вывода изготавливаются с небольшим количеством входов (обычно от 1 до 16), а сами модули располагаются вблизи места установки датчиков (рис. 2). Увеличение количества датчиков (входов) достигается путем наращивания числа модулей и объединения их с помощью общей шины. Это сокращает общую длину проводов в системе, а также длину проводов с аналоговыми сигналами.

Связь отдельных устройств в распределенной системе может осуществляться с помощью любой промышленной сети. Наиболее распространены в России сети Profibus, что связано с популярностью изделий фирмы Siemens, а также сети Modbus с физической шиной RS-485 благодаря распространенности модулей и контроллеров фирм ICP DAS, Advantech и НИЛ АП. За последние годы стремительно возросло количество используемых сетей Ethernet (точнее, Industrial Ethernet) в качестве промышленных сетей при скорости передачи 10, 100 и 1000 Мбит/с.

Распределенные системы строятся, как правило, из коммерчески доступных компонентов (ПЛК, модулей ввода-вывода, датчиков, исполнительных устройств). Однако для однотипных тиражируемых систем может быть выгодно строить специализированные системы, состоящие из полностью заказных (вновь спроектированных) аппаратных и программных средств. Граница целесообразности такого подхода определяется объемом

выпуска изделий. Программирование распределенных систем автоматизации выполняется стандартными средствами, рассмотренными далее.

Таким образом, **характеристики распределенной системы автоматизации и управления, отличающие ее от сосредоточенной (достоинства РСУ)** следующие:

- большее быстродействие благодаря распределению задач между параллельно работающими процессорами;
- повышенную надежность (отказ одного из контролеров не влияет на работоспособность других);
- большую устойчивость к сбоям (в силу простоты элементов и возможности резервирования);
- более простое наращивание или реконфигурирование системы;
- упрощенную процедуру модернизации;
- большую простоту проектирования, настройки, диагностики и обслуживания благодаря соответствию архитектуры системы архитектуре объекта управления, а также относительной простоте каждого из элементов системы и не зависимости от других элементов;
- улучшенную помехоустойчивость и точность благодаря уменьшению длины линий передачи аналоговых сигналов от датчиков к устройствам ввода и использованию цифровых интерфейсов передачи данных;
- меньший объем кабельной продукции, пониженные требования к кабелю и более низкая его стоимость;
- меньшие расходы на монтаж и обслуживание кабельного хозяйства.

Проектирование распределенных КСА

Для эффективного проектирования распределенных компьютерных систем автоматизации, обеспечения совместимости и взаимозаменяемости между собой всех устройств, входящих в систему и выпускаемых разными производителями необходимы строгие методы их описания. Для этих целей был разработан международный стандарт МЭК 61499 «Функциональные блоки для промышленных систем управления». Он использует три уровня иерархии моделей при разработке распределенных систем: **модель системы, модель физических устройств и модель функциональных блоков**. Модели всех уровней в соответствии со стандартом представляются в виде функциональных блоков, которые описывают процесс передачи и обработки информации в системе.

Особенностью функциональных блоков стандарта МЭК 61499 является то, что они учитывают не только традиционное инициирование выполнения алгоритма с помощью тактирования или временного расписания, но и по признаку наступления некоторых событий (событийное управление). Событийное управление является более общим, а тактирование можно рассматривать как его частный случай, заключающийся в периодическом появлении одного и того же события (сигнала тактирования). Функциональные блоки (ФБ) могут быть использованы также для поддержания всего жизненного цикла системы, включая проектирование, изготовление, функционирование, валидацию и обслуживание.

Модель распределенной системы автоматизации в соответствии со стандартом МЭК 61499 может быть представлена как набор физических устройств (например, ПЛК), взаимодействующих между собой с помощью одной или нескольких промышленных сетей (рис. 2). Сети могут иметь иерархическую структуру. Функции, выполняемые системой автоматизации, моделируются с помощью программного приложения, которое может располагаться в одном устройстве (Приложение В) или распределяться между несколькими устройствами (Приложения А и Б). Например, приложение, выполняющее ПИД-регулирование, может располагаться в трех устройствах, из которых первое выполняет функцию ввода данных от датчиков (является модулем ввода), второе выполняет алгоритм регулирования, третье выполняет функцию вывода данных в

исполнительное устройство. Вторым примером может быть клиент-серверное приложение, в котором клиент расположен в одном устройстве, сервер – в другом.

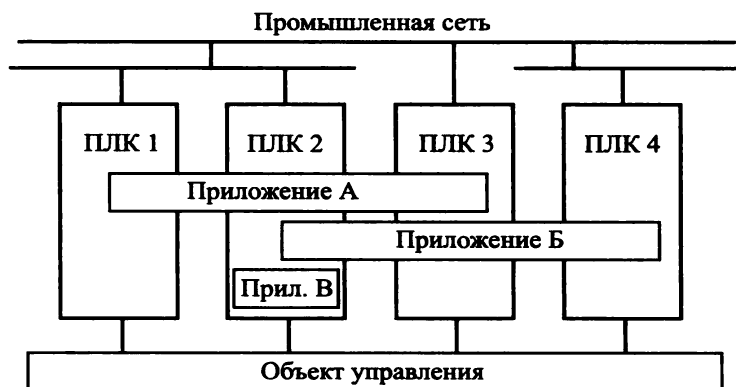


Рис. 2 – Модель РКСА по стандарту МЭК61499

Модель физического устройства. Каждое физическое устройство в распределенной системе должно содержать по крайней мере один интерфейс к объекту управления или к промышленной сети и может содержать несколько (в том числе ноль) ресурсов (рис. 3). При этом устройство по МЭК 61499 рассматривается как конкретный экземпляр определенного типа устройств. Интерфейс с объектом управления обеспечивает отображение данных и событий физического процесса (например, аналоговых или дискретных сигналов) в ресурсы и обратно.

В одном устройстве может быть несколько ресурсов и несколько программных приложений. Каждое приложение может исполняться на нескольких устройствах (рис. 2) и может занимать часть ресурсов в одном устройстве (рис. 3).

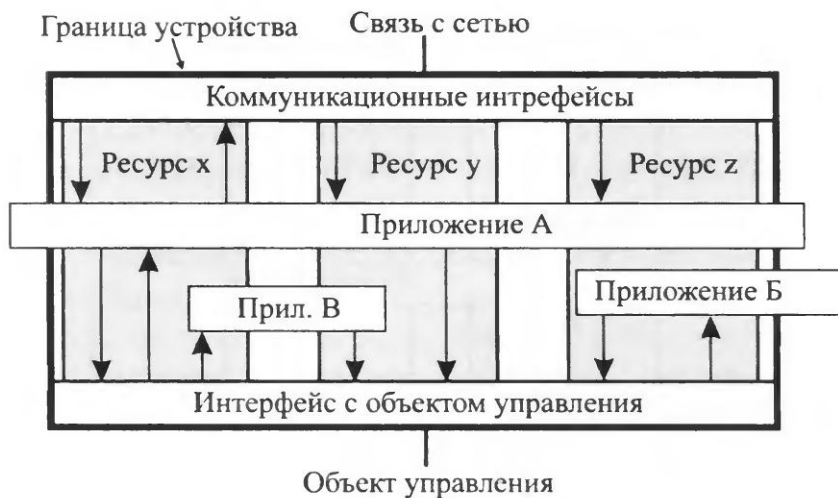


Рис. 3 – Пример модели устройства ПЛК2 по стандарту МЭК61499

Коммуникационные интерфейсы выполняют отображение между ресурсами и промышленной сетью. Они могут предоставлять информацию ресурсу в виде данных или событий, а также выполнять дополнительные функции для поддержки программирования, конфигурирования, диагностики и т.д.

Модель функциональных блоков образует модель приложения и модель ресурса.

Программное приложение состоит из сети функциональных блоков, ветви которой переносят данные и события (рис. 4). Поток событий определяет выполнение алгоритмов, содержащихся в функциональных блоках. В состав функциональных блоков могут входить и другие программные приложения (субприложения). Программные приложения могут быть распределены между несколькими ресурсами в одном или в нескольких устройствах (ПЛК).

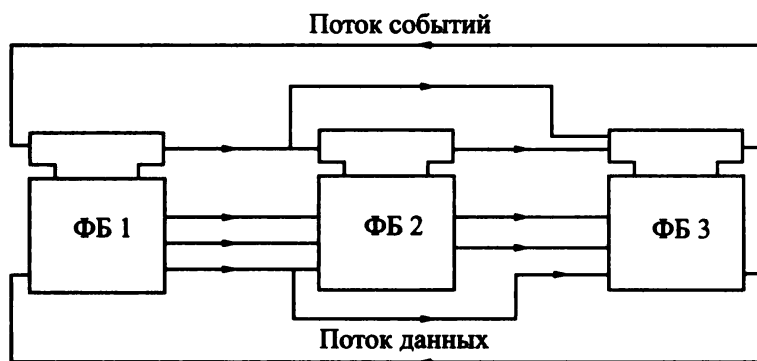


Рис. 4 – Модель приложения распределенной системы по стандарту МЭК61499

Ресурс рассматривается как *функциональная единица, которая содержится в устройстве* (например, в ПЛК), имеет независимое управление своими операциями и обеспечивает различные сервисные функции (сервисы) для программного приложения, включая планирование и исполнение алгоритмов. Ресурс может быть создан, сконфигурирован, параметризован, стартован, удален и т.п. без воздействия на другие ресурсы устройства. Примером ресурса может быть память и время, выделенные для выполнения задачи в центральном процессоре.

В функции ресурса входит прием данных или событий от объекта управления или коммуникационного интерфейса, обработка данных и событий и возврат данных и событий в процесс или промышленную сеть, в соответствии с алгоритмом работы программного приложения, использующего данный ресурс.

В соответствии с рис. 5, ресурс моделируется следующими элементами:

- локальным приложением, расположенным в данном физическом устройстве, или частью распределенного приложения. Данные и события, обрабатываемые в локальном приложении, принимаются со входов функциональных блоков (ФБ), которые выполняют операции, необходимые программному приложению;
- отображение событий и данных между программным приложением и объектом управления выполняется специальным функциональным блоком интерфейса (рис. 5);
- отображение событий и данных между программным приложением и коммуникационным интерфейсом выполняется таким же специальным функциональным блоком интерфейса (рис. 5);
- функция планирования выполняет передачу информации и исполнение программного приложения в соответствии с временными требованиями и очередностью, определяемой появлением событий, взаимодействием между функциональными блоками, а также приоритетами или периодом исполнения задач.

Ресурс реагирует на события, поступающие из интерфейсов, следующими способами:

- планированием и исполнением алгоритма;
- модифицированием переменных;
- генерацией ответных событий;
- взаимодействием с интерфейсами.

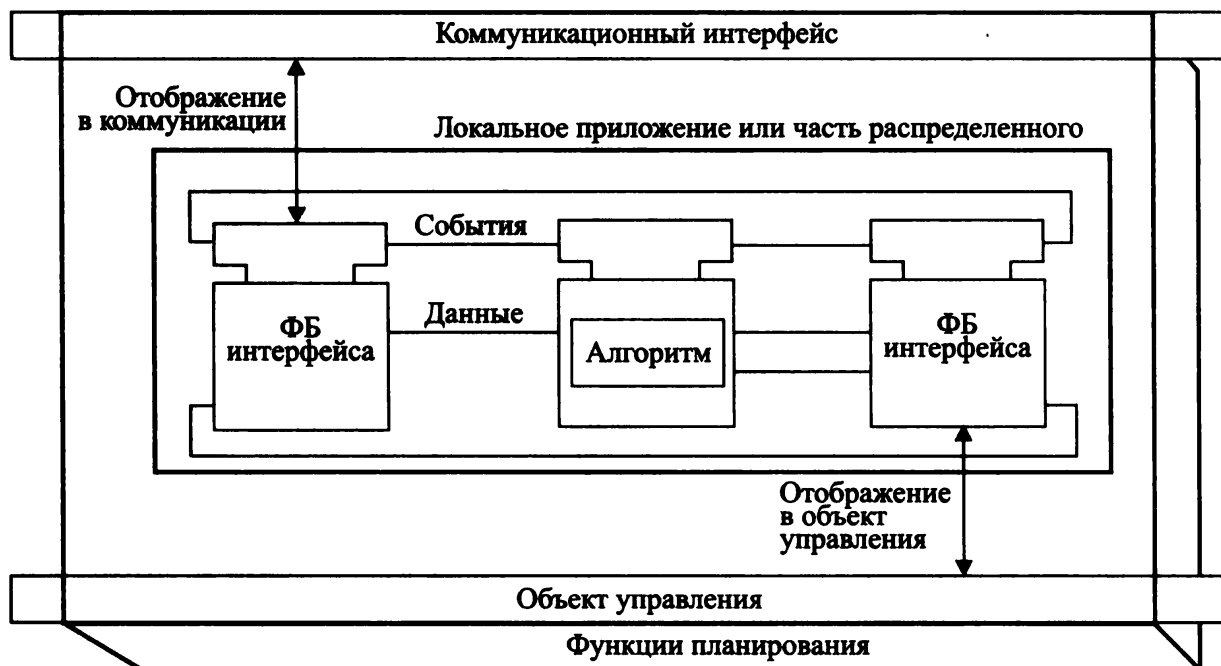


Рис. 5 – Модель ресурса распределенной системы по стандарту МЭК61499

На архитектуру и аппаратную реализацию распределенной КСА очень сильно влияют технические требования. **Основными техническими требованиями при проектировании распределенных КСА являются:**

- обеспечение широкого температурного диапазона работы технических средств локальных систем автоматического управления (САУ);
- распределенная система электропитания;
- обеспечение надежного контура заземлений на каждой отдельной площадке объекта автоматизации;
- защита контрольно-измерительных и информационных каналов от внешних воздействий, а также усиление передаваемых сигналов;
- выбор оптимального, с точки зрения эффективности, надежности и взаимозаменяемости составных частей, удовлетворяющего международным стандартам контроллерного оборудования;
- выбор оптимального, с точки зрения пылевлагодонепроницаемости, а также защиты от электромагнитного излучения, коррозии и др. факторов, удовлетворяющего международным стандартам конструктива шкафа цехового контроллера, шкафов автоматики локальных САУ и автоматизированного рабочего места системного инженера (АРМ);
- обеспечение высоконадежных каналов обмена технологической информацией между отдельными автоматизированными объектами и централизованной системой управления и контроля;
- резервирование основной аппаратуры контроля и управления, а также наиболее важных каналов передачи информации;
- обеспечение аппаратного и программного аварийного останова технологического комплекса при аварийных ситуациях;
- обеспечение высокоэффективного человеко-машинного интерфейса в системе визуализации и мониторинга;
- обеспечение обмена данными по информационным каналом в реальном масштабе времени;
- эффективная, с точки зрения скорости обнаружения неисправности, и надежная диагностика программно-аппаратных средств;

- обеспечение обслуживающего персонала качественной эксплуатационной документацией, а также инструментом для монтажа и диагностики.

Вопросы реализации данных требований рассматриваются далее в процессе ознакомления с аппаратной и программной реализацией уровней распределенной КСА.