

MASTERPLC DESIGNER — РОССИЙСКИЙ ИНСТРУМЕНТАРИЙ ДЛЯ РОССИЙСКИХ КОНТРОЛЛЕРОВ

ИЛЬЯ АБЛИН

ablin@insa.ru

MasterPLC Designer — новинка российского рынка промышленной автоматизации. Очевидным плюсом российского происхождения продукта является не только формальная пригодность для «импортозамещения», но и близость потребителя («контроллеростроителя») к разработчику продукта.

Независимых производителей систем программирования контроллеров во всем мире — единицы. Это связано с несоответствием между размером рынка и трудоемкостью полноценной реализации стандарта МЭК 61131-3. В то же время знание языков стандарта — FBD (схемы функциональных блоков), LD (релейно-контактная логика), SFC (схемы шаговых последовательностей) и текстового ST (структурированный текст) — стало обязательным требованием к инженеру по автоматизации, причем не только за рубежом, но и у нас. Официально в Россию поставляются программы ISaGRAF и CoDeSys, кроме того, в составе нескольких зарубежных контроллеров встречается MULTIPROG. Это все, что на слуху. Впрочем, ничего удивительного в этом нет, поскольку именно эти продукты и есть «три кита» мирового рынка. Других столь

же широко распространенных программ в этом сегменте практически нет. Эти среды программирования входят в комплект поставки большинства западных (и не только) контроллеров. Правда, производители первого ряда на них не ориентируются и для своих ПЛК выпускают собственный инструментарий (например, Siemens — Step7, Schneider Electric — Unity Pro, Rockwell Automation — RSLogic). Возможность программирования на языках МЭК 61131-3 предлагается и в немногочисленных SCADA, ориентированных на создание вертикально-интегрированных систем с единым проектом для верхнего уровня (сервер сбора данных и APM операторов) и нижнего (контроллеров с открытой архитектурой — SoftPLC). К числу таких программных продуктов «сквозного» проектирования относится и MasterSCADA производства компании «ИнСАТ». Но отдельной российской системы технологического программирования, независимой от производителей конкретных контроллеров, от SCADA и предназначенной для использования именно в качестве OEM-продукта для производителей или дистрибуторов ПЛК, а не для прямых поставок конечному пользователю, не было до тех пор, пока «ИнСАТ» не выпустил MasterPLC Designer. Очевидным плюсом российского происхождения этого продукта является не только формальная пригодность для «импортозамещения», но и близость потребителя к разработчику. А непосредственный кон-

такт с разработчиком необходим во многих случаях. Типичный пример: программный интерфейс расширения обычно не позволяет реализовать новую системную функцию, так как требуется более тесное взаимодействие с ядром исполнительной системы. Нередко встречается и желание компании — производителя контроллера выпустить продукт под своим брендом, со своим внешним видом и в собственной конфигурации предустановленных опций. Такие вопросы вряд ли успешно решаются в диалоге с иностранной компанией без чрезмерно высоких затрат или гарантий очень больших объемов закупок. А с «ИнСАТ» их решить можно.

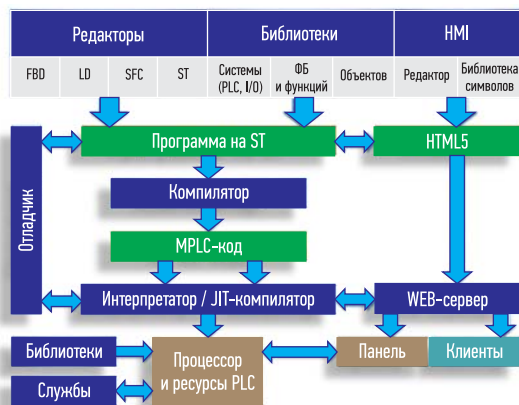
СТРУКТУРА И СОСТАВ ПРОДУКТА

Структура и состав MasterPLC Designer в целом типичны для продуктов этого класса, но, как и у всех, в нем есть некоторая своя собственная специфика (рис. 1 и 2).

MasterPLC Designer состоит из двух программ: инструментальной среды и исполнительной системы. Инструментальная среда — это Windows-приложение, а исполнительная система — кроссплатформенная программа, то есть она может работать практически на любой программно-аппаратной платформе (процессоры x86, x64, ARM и др., операционные системы — WindowsCE/Embedded, Linux и др.).

Разумеется, для каждой платформы, точнее даже для каждого типа поддер-

▼ РИС. 1. Создание в среде MasterPLC Designer программ контроллеров



живаемых контроллеров, существует специальная версия исполнительной системы, полученная из базового кода путем его компиляции совместно с набором специфичных для данной платформы и контроллера библиотек столь же специфичным для данной платформы компилятором. Соответственно, и название ее отражает эту специфику: от MasterPLC for Windows до MasterPLC for «ВашКонтроллер». Причем такая подготовка исполнительной системы к работе на конкретной модели контроллера доступна и для самостоятельного применения производителю контроллера, получившему от «ИнСАТа» комплект портирования. Адаптация внешнего вида инструментальной системы для создания OEM-версии с логотипом и дизайном заказчика выполняется под заказ и не требует программирования, так как заключается в создании нового стиля, описанного в наборе XML-файлов.

Программы, созданные с помощью инструментальной среды на графических языках FBD, SFC и LD, предварительно компилируются в ST с возможностью просмотра полученного текста в среде программирования. Этот текст, как и программы, изначально написанный на языке ST, компилируется в промежуточный (или целевой) код, предназначенный для загрузки в исполнительную систему контроллера. В зависимости от конкретной исполнительной системы полученный код может в ней интерпретироваться с помощью встроенного интерпрета-

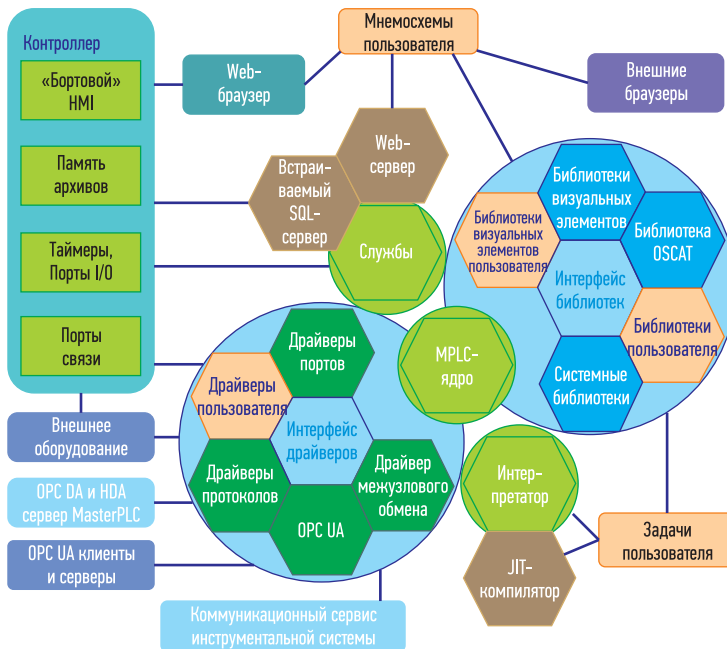


Рис. 2. Состав и архитектура исполнительной системы MasterPLC

тора либо непосредственно перед первым исполнением компилироваться в код целевой платформы с помощью JIT-компилятора (компиляция Just In Time — «на лету»).

Существенным отличием MasterPLC Designer от аналогов является подход к организации HMI — визуализации «на борту» контроллера. В конкурирующих продуктах этот функционал существенно уступает аналогичным возможностям SCADA-систем либо просто отсутствует. В MasterPLC Designer встроен графический редак-

тор с обширными библиотеками и возможностью просмотра окон мнемосхем, трендов и журналов сообщений не только на собственной панели контроллера, но и через Web-доступ на любом клиентском устройстве. Подробнее мы рассмотрим это далее.

ИНТЕГРИРОВАННАЯ СРЕДА

Интегрированная среда MasterPLC Designer (рис. 3) реализована с учетом многолетнего опыта эксплуатации MasterSCADA, оболочка которой

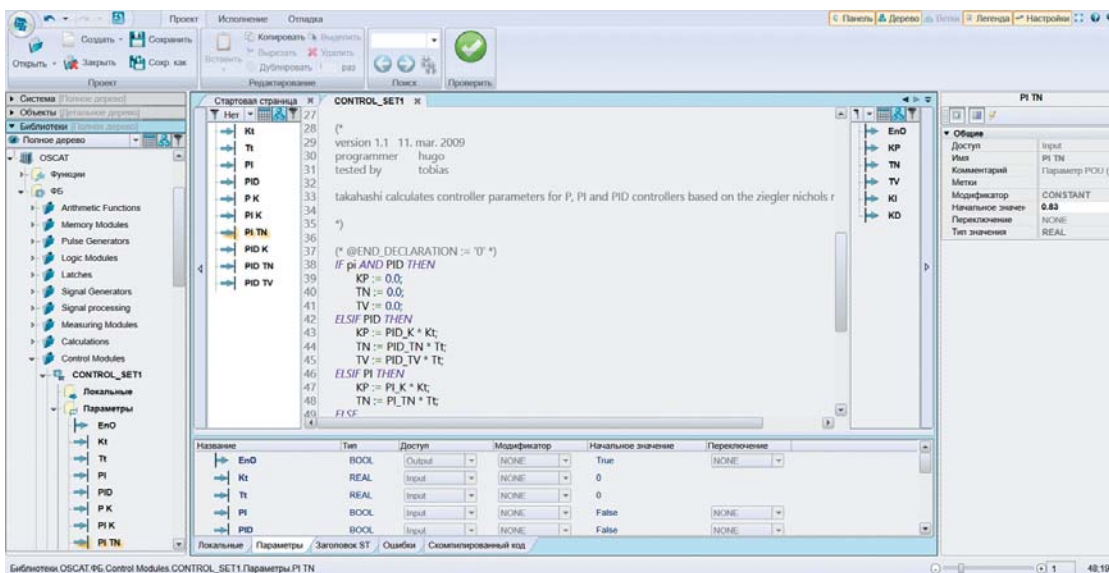
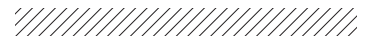


Рис. 3. Интегрированная среда MasterPLC Designer



стоит на двух «китах»: объектном подходе с разделением физической реализации системы и абстрактной модели объекта и принципе построения среды «все в одном», когда для каждого элемента проекта нужны ему редакторы или свойства открываются на закладках.

Устройство интегрированной среды MasterPLC Designer осталось тем же: это организация проекта в виде двух объектных иерархий: дерева системы (рис. 4а) и дерева объекта (рис. 4б). Дерево системы содержит контроллеры, поддерживаемые ими протоколы связи и службы, подключенные модули и их сигналы. Поскольку программы могут разрабатываться и непосредственно для контроллера, и для любого его встроенного или внешнего модуля, в проекте можно обойтись и одним деревом системы. Однако правильнее использовать эту возможность только для первичной обработки сигналов, специфичной именно для данного типа оборудования, а логику контроля и управления разрабатывать в аппаратно-независимом дереве объектов, что позволяет в любой

момент изменить типы и состав оборудования без переделки проекта. При таком подходе можно весь проект разработать без привязки к «железу», а лишь затем включить в дерево системы его описания и установить связи между двумя иерархиями. В дополнение к традиции в состав проекта было включено дерево подключенных библиотек, что повысило удобство работы с ними, позволило создавать проектные библиотеки в том же виде, что и сам проект. Разработка проекта состоит в создании иерархий, связей между их элементами, программ обработки, сообщений и окон (для контроллеров с локальной или веб-визуализацией). Все действия производятся, в основном, методом «перетаски и брось». Для удобства работы и установления связей любую ветку дерева можно открыть в отдельном окне.

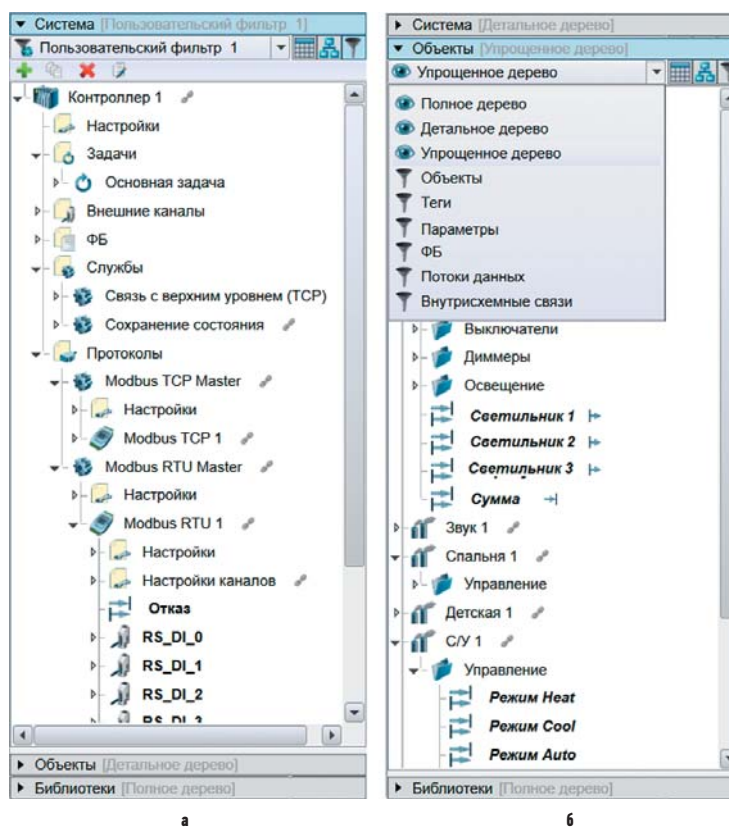
В окне программы появились также панели «легенды» (для работы с элементами программы открытого редактора) и таблицы свойств выбранного элемента (рис. 3). Пользователь может гибко управлять внешним видом сре-

ды разработки: любая из панелей может быть отключена в случае ненужности кнопкой, расположенной в строке меню программы, а содержимое каждой из панелей может быть индивидуально отмасштабировано для удобства работы. Детальная справка с выбором одной из потенциально возможных для данного контекста тем из меню позволяет пользователю «по месту» уточнить все неясности.

Опыт общения с пользователями научил специалистов «ИнСАТ» многообразию их потребностей и уровня подготовки. Пользователи должны иметь возможность подстроить представление проекта под свои индивидуальные запросы. Этим целям служат фильтры, позволяющие убрать из проекта ненужные детали (рис. 4б). Наряду с предопределенными фильтрами возможно создавать и сохранять свои собственные, любой сложности, с отбором по именам, свойствам, отношениям, пользовательским меткам и др. Например, можно оставить из всего проекта только все температуры воды. Безусловно, в этом случае удобнее использовать табличное представление списка параметров по алфавиту, для переключения на которое имеется кнопка в титульной строке каждого дерева.

СОСТАВ ПРОЕКТА

Мы уже упоминали кратко состав дерева системы. Его создание идет сверху вниз. Добавляются контроллеры, в них добавляются встроенные модули либо протоколы связи, назначенные на конкретные порты, и в них — модули удаленного ввода/вывода. Модули имеют свой набор каналов. Канал ввода/вывода — это набор из типовых параметров: значение, метка времени, статус опроса. Состав модулей и типы каналов предопределены разработчиком библиотеки, но пользователь может создавать и собственные типы модулей и каналов, в том числе со встроенной обработкой сигналов. К примеру, в Modbus-устройстве он может разбить адресное пространство на такой набор модулей и каналов, который соответствует физической структуре устройства. Контроллерам, модулям и каналам дерева системы могут принадлежать не только системные про-



► РИС. 4.
а) Дерево системы;
б) дерево объекта

граммы (драйверы), включенные в состав конкретной исполнительной системы, но и любые пользовательские программы, созданные для конкретного проекта, либо библиотеки.

Иерархия дерева объектов состоит из объектов, тегов и параметров. Объектам и тегам так же, как и перечисленным выше элементам дерева системы, могут принадлежать ресурсы: программы, окна, сообщения и журналы сообщений. В отличие от них параметры — это атомарные элементы проекта, в которые нельзя ничего включить и которые не имеют собственных программ или окон. Создание типа параметра производится в специальном диалоге выбором из простого, библиотечного или сложного (массив, структура) типа данных (рис. 5).

БИБЛИОТЕКИ

МЭК 61131-3 предусматривает обязательное наличие стандартной библиотеки базовых функций и функциональных блоков. Для крупноблочного построения программ этого набора элементарных алгоритмов, конечно, недостаточно. Пользователь может наращивать собственные библиотеки (а таким образом поступают все профессиональные разработчики), однако никому не хочется заниматься изобретением велосипеда. Международное сообщество инженеров по автоматизации уже несколько лет назад создало и продолжает развивать библиотеку OSCAT из свыше 500 функций и функциональных блоков с открытыми текстами на языке ST. Однако из-за неполной совместимости реализации стандарта в разных системах программирования библиотека существует в нескольких версиях: для Step7, для CoDeSys и др. Выполненная «ИнСАТ» реализация практически не отклоняется от определенной стандарта, что потребовало очищения кода библиотеки от нестандартных приемов, например использования указателей. Существенное отличие реализации библиотеки OSCAT заключается еще и в том, что включенный в MasterPLC Designer вариант не требует ручного переноса кода ST в собственные программы: библиотека заранее целиком вклю-

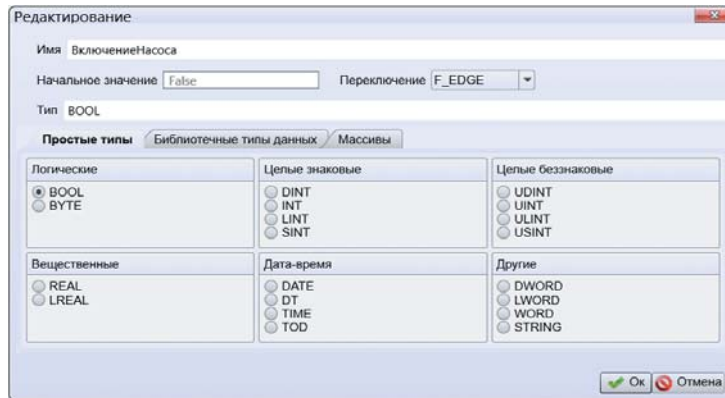


Рис. 5. Создание параметра

чена в среду продукта, полностью адаптирована, протестирована и снабжена документацией на русском языке.

Наряду с этими базовыми библиотеками в составе MasterPLC Designer поставляются библиотеки оборудования, типов данных, типов тегов и каналов.

РЕДАКТОРЫ ТЕХНОЛОГИЧЕСКИХ ЯЗЫКОВ

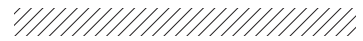
MasterPLC Designer построен так, чтобы в любой момент можно было добавить новые редакторы, ничего не меняя в самой среде. Эта предсказуемость связана не только с нашими собственными планами, но и с потенциальными перспективами развития стандарта, который давно уже засиделся в четырех стенах существующих языков.

Ранее было отмечено, как выглядит редактор текстового языка ST (рис. 3), который обладает всеми стандартными возможностями таких редакторов (контекстным помощником ввода и раскраской ключевых слов, отладчиком и др.). Но на один момент все же хотелось бы обратить внимание читателя. Это списки входных параметров программы слева и выходных параметров справа от поля самого редактора. Так выглядит механизм программных клеммников. Он позволяет разработчику, использующему готовую программу, работать с ней, даже если ее код закрыт от просмотра. Но и в случае, если он открыт, механизм клеммников обеспечивает промежуточный слой, который защищает программу от неосторожного вмешательства, позволяет легко

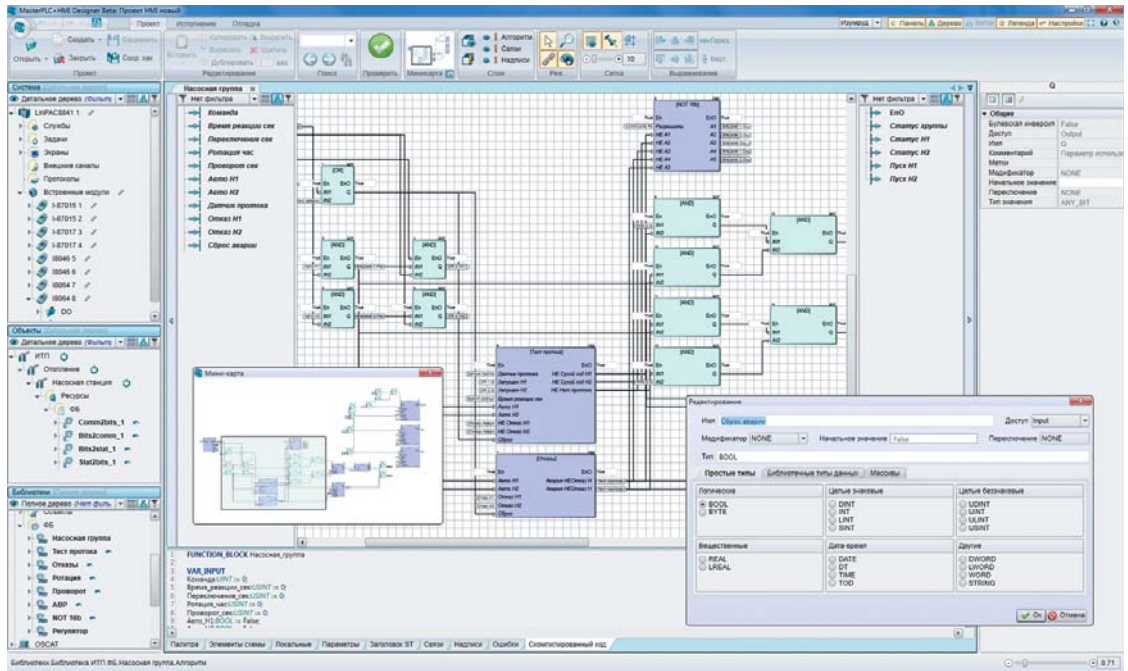
переносить ее в библиотеку или брать оттуда без коррекции кода, помогает пользоваться готовым кодом, не вникая в его внутреннее устройство. Число переменных в клеммнике большой программы может быть велико, поэтому к этому списку можно применять те же инструменты фильтрации, что и к основному проекту, с той разницей, что для клеммника изначально используется табличное представление параметров.

Редактор схем функциональных блоков (рис. 6) содержит большое число сервисных функций, обеспечивающих удобство «рисования» алгоритмов контроля и управления. На схему можно перетащить не только библиотечный блок, но и программу на любом языке, клеммники которой при этом преобразуются во входы/выходы блока на схеме. Автоматическая прокладка линий связи с возможностью их ручной корректировки, привязка к сетке при расстановке блоков, надписи, управление видимостью слоев, расстановка значений в режиме отладки прямо на входах/выходах блоков — это далеко не полный список имеющихся сервисных возможностей.

Отдельно стоит упомянуть окно мини-карты, которое позволяет перемещать видимую зону на склоне изображении всей схемы в том случае, когда она слишком велика, чтобы поместиться на экран. Этот механизм работает и в других графических редакторах, например редакторе языка SFC (рис. 7), предназначенного для управления последовательностью выполнения программ, написанных на иных языках стандарта.



► РИС. 6.
Редактор схем функциональных блоков



НМИ — ВИЗУАЛИЗАЦИЯ «НА БОРТУ» И В ОБЛАКАХ

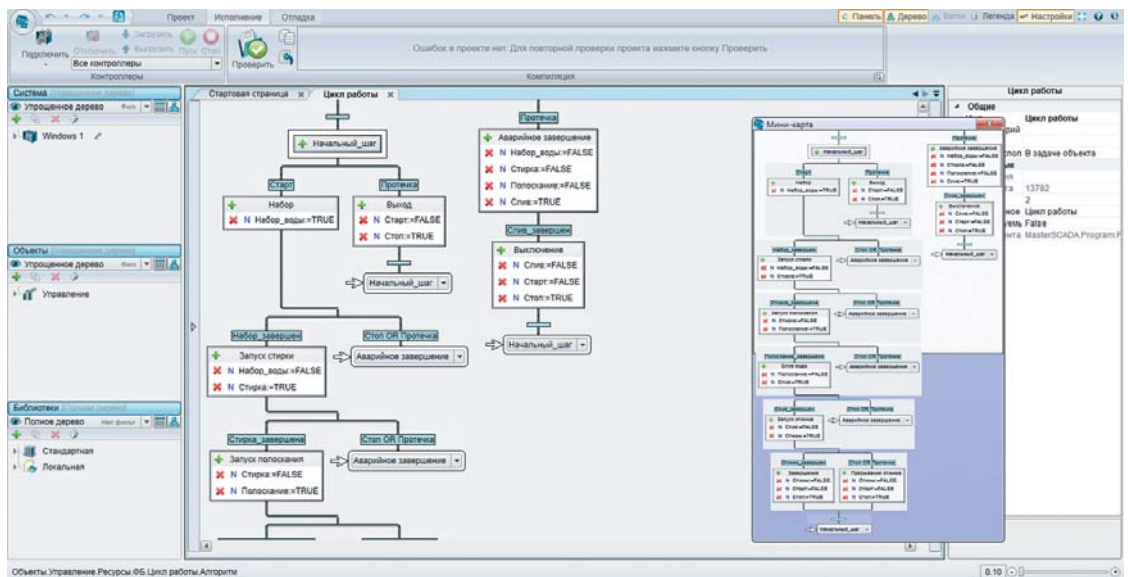
Новые времена — новые возможности контроллерных процессоров и программ. Сейчас уже стало нормой, что контроллер имеет встроенный веб-сервер. Технологически эту возможность предоставляют современные операционные системы, устанавливаемые на контроллеры, прежде всего Linux. Однако мало у каких контроллеров наличие технологических предпосылок приводит

к появлению полноценной графики на уровне современных SCADA-систем. В MasterPLC Designer реализован полноценный редактор полностью векторных на базе формата SVG HTML-страниц, укомплектованный общепромышленной библиотекой проектных объектов типа задвижки, насоса и т. п., а также рядом отраслевых библиотек. Эти объекты наряду с логикой контроля и управления, реализованной в программе на ST, имеют и отображение в виде анима-

рованных 3D-символов и окон управления (рис. 8).

Возможности редактора не ограничиваются использованием готовых объектов. Пользователь может создать из базовых примитивов новый символ, любое свойство которого может быть динамизировано параметрами проекта. При этом обеспечивается верстка с использованием любых типов панелей для организации нужного расположения элементов, а готовая мнемосхема име-

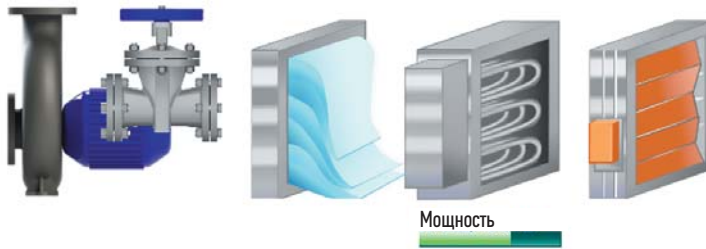
► РИС. 7.
Редактор SFC и МЕХАНИЗМ МИНИ-КАРТЫ



ет ту степень адаптации к размеру экрана, которую заложил разработчик проекта. Например, «резиновая» (то есть растягивающаяся под размер клиентского экрана) верстка может содержать элементы с ограниченным минимальным размером (текст, изображения и т. п.). В результате одна и та же мнемосхема (рис. 9) с равным успехом может отображаться на встроенной панели контроллера, подключенном непосредственно к нему мониторе или на клиентском устройстве (компьютере, планшете, смартфоне), подключенном через Интернет. Примеры таких мнемосхем можно увидеть на сайте www.sky-monitor.ru.

MASTERPLC DESIGNER — ПОЛНОЦЕННЫЙ ФУНКЦИОНАЛ ДЛЯ СЕРЬЕЗНЫХ ПРИМЕНЕНИЙ

Современный контроллер — это не просто устройство для опроса сигналов датчиков и выработки управляющих воздействий. Одной из важнейших его функций является архивирование изменений значений параметров и регистрация событий. Исполнительная система MasterPLC обеспечивает не просто сбор этих данных и передачу их по каналам связи на верхний уровень системы, но и отображение графиков изменений параметров и журналов сообщений во встроенных средствах визуализации.



◀ РИС. 8. Библиотеки анимированных объектов

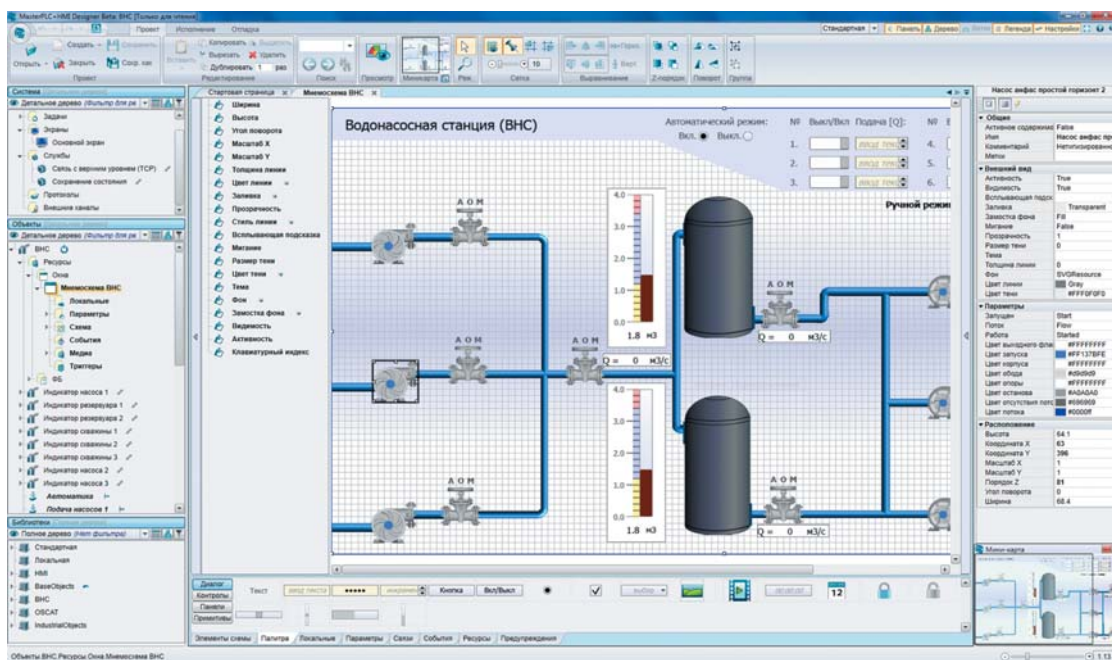
OPC UA — ОСНОВА КРОССПЛАТФОРМЕННЫХ КОММУНИКАЦИЙ

Широко распространенные протоколы связи с контроллером, например Modbus, имеют существенный недостаток. Они не позволяют обратиться к контроллеру через маршрутизируемую сеть, например через Интернет. Не спасают положение и классические OPC-серверы, установленные на находящемся в той же локальной сети компьютере. Клиентское приложение также не может обратиться к ним из внешней сети без использования дополнительного канала-образующего ПО. Выходом является наличие сервера кроссплатформенного стандарта OPC UA непосредственно на борту контроллера. Стандарт OPC UA не только обеспечивает транспорт данных в гетерогенных IP-сетях, но и обеспечивает сопровождение данных из контроллера их описанием, что позволяет подключать к контроллеру

новых клиентов «на ходу», без передачи им конфигурационных файлов. Включение в состав исполнительной системы MasterPLC OPC UA сервера существенно расширяет спектр возможных применений данного продукта, в том числе и его использование в качестве серверного ПО сбора и отображения данных на любых клиентских устройствах через Интернет.

MASTERPLC DESIGNER — ПРОДОЛЖЕНИЕ СЛЕДУЕТ...

Читатель, вероятно, заметил, что описанный функционал позволяет рассматривать MasterPLC Designer не только как систему программирования контроллеров. Действительно, возможны и иные применения, полностью стирающие границы традиционной классификации элементов системы автоматизации и меняющие как ее архитектуру, так и способы работы с ней. Впрочем, об этом мы расскажем в следующей статье. ●



◀ РИС. 9. Мнемосхема насосной станции