



ПРОГРАММНО-ТЕХНИЧЕСКИЙ КОМПЛЕКС «СУРА»

Фирменное программное обеспечение
«Мегаполис»

Описание программы

АДИГ.00052-01 13 01

Листов 21

Оглавление

1	Введение.....	4
2	Мегapolis. Назначение и условия применения	5
2.1	Терминология.....	5
3	Общие сведения о Мегapolisе.....	7
3.1	Структура проекта	7
3.2	Основные функции Мегapolisа	7
3.3	Дополнительные функции Мегapolisа.....	7
3.3.1	Компиляция и загрузка	7
3.3.2	Обзор выполнения технологической программы.....	7
3.3.3	Диагностика и настройка.....	8
3.3.4	Блокировка	8
3.3.5	Перестановка.....	8
4	Работа с Мегapolisом.....	9
4.1	Главное окно Мегapolisа.....	9
4.2	Шкала инструментов.....	9
4.3	Меню Вид.....	10
4.3.1	Состав меню вид.....	10
4.3.2	Результаты поиска.....	10
4.3.3	Обзор проекта	10
4.3.4	Объекты.....	10
4.3.5	Выбор алгоритма	11
4.3.6	Порядок выполнения.....	11
4.3.7	Список ошибок	11
4.4	Свойства задач	11
4.4.1	Список задач	11
4.4.2	Поле задачи	11
4.4.3	Размещение алгоблоков	12
4.4.4	Реквизиты алгоблоков.....	12
4.4.5	Входы-выходы алгоблока.....	12
4.4.6	Копирование алгоблоков	13
4.4.7	Свойства входов алгоблоков	13
4.4.8	Общий случай установки констант.....	13
4.4.9	Выбор типа данных	13
4.5	Конфигурирование алгоблоков	13
4.5.1	Техника конфигурирования.....	13
4.5.2	Ссылки.....	14
4.6	Перемещение и перестановка алгоблоков.....	14
4.6.1	Возможности перемещения	14
4.6.2	Наложение алгоблоков и ссылок	14
4.6.3	Перестановка алгоблоков	14

4.7	Комментарии.....	14
4.7.1	Виды комментариев	14
4.7.2	Комментарии к задаче.....	14
4.7.3	Другие виды комментариев.....	15
4.8	Привязка	15
4.8.1	Назначение привязки	15
4.8.2	Отображение привязки	15
4.9	Права доступа и многопользовательская работа с Мегаполисом	15
4.9.1	Права доступа	15
4.9.2	Многопользовательская работа.....	15
5	Отладка техпрограмм	17
5.1	Компиляция и загрузка техпрограммы.....	17
5.2	Безопасная загрузка	17
5.3	Обзор.....	18
5.3.1	Возможности обзора	18
6	Рекомендации по подготовке техпрограмм	19
6.1	Ввод/вывод информации через модули УСО	19
6.2	Межконтроллерный обмен	19
6.3	Объектные алгоритмы.....	19
6.3.1	Информация об объектах.....	19
6.3.2	Общие свойства объектных алгоритмов	19
6.4	Вычислительные алгоритмы	19
6.4.1	Состав вычислительных алгоритмов.....	19
6.5	Архивирование сигналов	20
6.5.1	Варианты архивирования	20
6.5.2	Периодическое архивирование сигналов.....	20
6.5.3	Архивирование аналоговых сигналов по апертуре.....	20
6.5.4	Объектное архивирование	20
	Лист регистрации изменений	21

1 Введение

Содержание данного документа может изменяться без предварительного уведомления.

В связи с непрерывной работой по улучшению характеристик изделия, технические данные могут изменяться. Тем не менее, содержание документации регулярно отслеживается, и все корректировки вносятся в последующие издания.

Настоящее руководство описывает фирменное программное обеспечение «Мегаполис», входящее в состав ПО ПТК «СУРА», содержит сведения о его характеристиках, а также другие сведения, необходимые для правильной эксплуатации.

Мегаполис – это IDE (интегрированная среда разработки) – фирменное программное обеспечение верхнего уровня ПТК «СУРА», предназначенное для подготовки прикладных программ для контроллеров, компиляции и загрузки подготовленной программы в контроллер, для описания и задания параметров и объектов в проекте, а также для выполнения других вспомогательных функций, необходимых для подготовки проекта АСУ ТП к работе.

2 Мегаполис. Назначение и условия применения

Система технологического программирования Мегаполис предназначена для подготовки пользовательских технологических программ фирменных универсальных проектно-компонруемых контроллеров Elicont-100, Elicont-200 (далее по тексту – контроллер), входящих в программно-технический комплекс «СУРА» (далее – ПТК «СУРА»), а также для редактирования базы данных объектов и их свойств. В Мегаполисе осуществлена поддержка языков технологического программирования FBD и ST стандарта МЭК 61131.

При работе с Мегаполисом используется библиотека алгоритмов (более 200 библиотечных модулей), описание которой «встроено» в Мегаполис и может быть вызвано на экран монитора станции проектирования.

2.1 Терминология

Ниже перечислены основные термины, используемые в тексте и специфичные для Мегаполиса.

База данных (БД) – совокупность всех проектных данных об АСУ ТП, формируемая пользователем и необходимая для работы СУРЫ. В состав БД входят и программы, созданные средствами Мегаполиса.

Технологическая программа – совокупность формируемых пользователем данных (задач, алгоблоков, конфигурационных связей между ними и т.д.), загружаемых в контроллер.

Информационный объект (в дальнейшем – объект) – совокупность информации, характеризующая свойства информационного элемента АСУ ТП как единого целого. Информационные объекты могут отражать свойства реальных (датчики, механизмы и т.п.) или виртуальных (регуляторы, шаговые программы и т.п.) элементов.

Задача – совокупность алгоблоков, имеющая имя и размещаемая в одном непрерывном поле задачи. Обычно задача Мегаполиса соответствует сравнительно автономной технологической задаче управления.

Подзадача – задача, иерархически входящая в другую задачу.

Поле задачи – непрерывная область размещения алгоблоков, которая, в частности, может быть больше экрана компьютера. В последнем случае для просмотра алгоблоков, входящих в задачу, используется прокрутка поля задачи.

Алгоблок – алгоритм, введенный в поле задачи.

Алгоритм – созданная разработчиками СУРЫ фиксированная совокупность функций, имеющая имя и входящая в библиотеку алгоритмов контроллера.

Объектный алгоритм – алгоритм, формирующий совокупную информацию об информационном объекте.

Конфигурация – система связей между алгоблоками.

Свободный вход – не законфигурированный (т.е. ни с чем не связанный) вход алгоблока, на котором задается константа.

Константа – значение сигнала на свободном входе.

Связанный вход – законфигурированный вход алгоблока, т.е. вход, соединенный с каким-либо выходом какого-либо алгоблока.

Ссылка – мнемоническое обозначение адреса источника или приемника, заменяющее линию связи между алгоблоками.

Перестановка – изменение порядка выполнения задач в пределах контроллера или алгоритмов в пределах задачи.

Привязка – связь между алгоблоками или отдельными его элементами (каналами, входами, выходами) с объектами базы данных.

Компиляция – процесс преобразования подготовленной в Мегаполисе технологической программы в специальный формат, используемый для загрузки этой программы в контроллер.

Обзор – наблюдение средствами Мегаполиса за реальными сигналами, формируемыми контроллером на входах и выходах алгоблоков. Обзор используется для отладки подготовленной технологической программы.

3 Общие сведения о Мегаполисе

3.1 Структура проекта

АСУ ТП может содержать сотни и даже тысячи различных компонент, и чтобы в них можно было легко ориентироваться, им необходимо придать определенную иерархическую структуру. В СУРЕ можно создавать в большой степени произвольную структуру, но лучше всего, если она будет соответствовать внутренней связи отдельных компонент технологического процесса.

Все логические элементы АСУ ТП, связанные с контроллерами, структурируются в СУРЕ следующим образом:

Общие управляющие функции распределяются по контроллерам. Каждому контроллеру присваивается технологическое имя и сетевой номер.

Функции каждого контроллера разбиваются на задачи. Задачам даются технологические имена.

Каждая задача может разбиваться на подзадачи, каждая подзадача – на свои подзадачи и т.д. Иерархия и глубина вложенности задач и подзадач определяется пользователем.

Состав и структура задач и подзадач – непосредственно в Мегаполисе.

Свойства задач и подзадач в большинстве своем совпадают, поэтому в дальнейшем при их описании будет использоваться общий термин – задача. Термин подзадача будет использоваться только там, где нужно подчеркнуть соподчиненность этих понятий.

3.2 Основные функции Мегаполиса

После того, как определена структура проекта, можно приступать к подготовке технологических программ. В процессе этой подготовки в Мегаполисе выполняются следующие действия:

- выбирается требуемый контроллер;
- для выбранного контроллера создается перечень задач. Выбирается требуемая задача, - на экране открывается поле задачи;
- из библиотеки алгоритмов контроллера выбираются требуемые алгоритмы, и они расставляются наглядным образом в поле задачи по указанию пользователя;
- между входами и выходами отдельных алгоритмов проводятся требуемые связи;
- на свободных входах алгоритмов задаются требуемые константы;
- некоторые элементы полученной алгоритмической структуры привязываются к объектам БД. Такая привязка необходима для отображения информации, формируемой контроллерами, на АРМ.

Таким образом, технологическая программа готовится для конкретного выбранного контроллера.

3.3 Дополнительные функции Мегаполиса

3.3.1 Компиляция и загрузка

После того, как контроллер подключился к информационному пространству, подготовленную для него средствами Мегаполиса пользовательскую технологическую программу можно откомпилировать и в контроллер загрузить.

В процессе компиляции технологическая программа, имеющая вид функциональной схемы, преобразуется в специальный формат, используемый для загрузки этой программы в контроллер.

В одной процедуре загрузки программа загружается в один выбранный контроллер, при этом загружаются все задачи технологической программы.

3.3.2 Обзор выполнения технологической программы

Мегаполис предоставляет средства для отладки, загруженной в Контроллер программы, при этом могут использоваться модели объекта или (в более простом случае) имитаторы сигналов.

В специальном режиме, называемым Обзор, непосредственно в Мегаполисе можно наблюдать реальные значения сигналов на входах и выходах алгоблоков. Изменяя сигналы с помощью имитаторов или изменяя значения констант на входах алгоритмов, можно наблюдать реакцию на выходах выбранных алгоритмов и делать соответствующие выводы.

Из Мегаполиса можно послать в контроллер объектную команду, имитирующую действие оператора. Например, можно перевести регулятор на ручной или автоматический режим, включить или отключить двигатель, открыть или закрыть задвижку и т.д. Результат посланной команды можно наблюдать по изменению сигналов на входах-выходах, выбранных алгоблоков или на экране операторской станции.

Функции обзора работают, только если рабочая станция с Мегаполисом подключена к общей сети, к которой подключен соответствующий контроллер с загруженной технологической программой.

3.3.3 Диагностика и настройка

Для выбранного контроллера Мегаполис позволяет посмотреть использованные технологической программой ресурсы, запросить приборные параметры контроллера и сравнить их с данными БД, посмотреть содержимое буфера ошибок, как результат работы системы встроенной самодиагностики.

3.3.4 Блокировка

При коллективной работе над проектом могут возникнуть информационные конфликты между различными пользователями. Например, два пользователя могут открыть одну и ту же задачу, при этом один из них будет «вмешиваться» в работу другого. Другой пример – один пользователь начал загружать программу в контроллер, в то время как другой продолжает ее редактировать.

Чтобы исключить подобные конфликты, которые могут нарушить целостность базы данных и привести к некорректной работе программы, в Мегаполисе предусмотрен механизм блокировок, который, в зависимости от ситуации, распространяется на весь контроллер или отдельные входящие в него задачи.

3.3.5 Перестановка

Для работы в реальном времени важен не только состав и конфигурация задач и алгоритмов, но и последовательность их выполнения. После того, как подготовленная технологическая программа будет загружена в контроллер, сначала выполнятся все алгоритмы задачи, стоящей в Мегаполисе выше по списку, затем все алгоритмы первой по списку подзадачи, входящей в данную задачу и т.д. В пределах каждой задачи алгоблоки выполняются в той последовательности, в которой они вводились в Мегаполисе.

Если текущий порядок задач или алгоблоков не соответствует необходимому, его можно изменить, воспользовавшись процедурой Перестановка. Данная процедура позволяет переназначить порядок выполнения уже введенных задач и алгоблоков.

4 Работа с Мегаполисом

4.1 Главное окно Мегаполиса

После запуска приложения открывается главное окно Мегаполиса.

В меню вид главного окна Мегаполиса можно задать конфигурацию панелей приложения. В шапке окна указывается название приложения, путь (локальный или сетевой) к БД и имя выбранного контроллера. Основное место главного окна Мегаполиса занимает поле задачи с алгоблоками. Одновременно можно открыть несколько задач и расположить в различных вариациях при помощи панели привязки.

Центральное расположение основное и единственное для первой открытой задачи. Остальные задачи можно располагать в удобном для пользователя положении. По умолчанию все задачи открываются с центральной привязкой в виде вкладок в главном окне.

4.2 Шкала инструментов

В базовом окне мегаполиса доступны следующие инструменты:

Заккрыть все вкладки. Закрываются все открытые ранее вкладки в поле задач.

Сохранить. Сохранить макрос на текстовом языке ST.

Вырезать. Вырезать один или несколько выделенных алгоблоков в поле задач.

Копировать. Копировать один или несколько алгоблоков в поле задач, в другую задачу текущего контроллера, в задачу другого контроллера.

Вставить. Вставка одного или нескольких алгоблоков в поле задач, в другую задачу текущего контроллера, в задачу другого контроллера.

Отменить. Отменяет последнее действие пользователя при редактировании техпрограммы. В случае если надо отменить несколько последних изменений техпрограммы, то данная кнопка нажимается необходимое количество раз.

Повторить. Восстанавливает при редактировании техпрограммы последнее отмененное действие пользователя.

Обновить. Обновить (актуализировать) информацию.

Выделить все. Выделить все элементы (алгоблоки, связи) в поле задачи или макроса.

Проверить код техпрограммы на наличие ошибок. Проверяет код техпрограммы на наличие ошибок.

Войти в режим обзора. Запускает режим Обзора.

Выйти из режима обзора. Прекращает работу обзора.

Увеличить масштаб. Увеличить масштаб изображения в поле задач.

Уменьшить масштаб. Уменьшить масштаб изображения в поле задач.

Установить масштаб 1:1. Устанавливается 1:1, если ранее был изменен.

Настройка типа алгоблока. Выбор типа данных.

Увеличение модификатора. Увеличивает модификатор для выбранного алгоблока.

Уменьшение модификатора. Уменьшает модификатор для выбранного алгоблока.

Задать значение модификатора. Окно для задания значения.

Включить качество. Включает качество сигналов.

Отключить качество. Отключает качество.

Инверсия. Добавляет знак инверсии на входе алгоблока.

Без инверсии. Удаляет знак инверсии на входе алгоблока.

Заменить на ссылки. Заменяет линию связи, между алгоблоками, на ссылки.

Заменить на линию. Ссылки на соответствующих входах и выходах алгоблоков заменяются на линию.

Добавить комментарий. Добавляет комментарий на поле задачи или макроса.

Редактировать комментарий. Редактирование комментариев.

Заблокировать. Запрещает редактировать задачу другим пользователям.

Разблокировать. Разрешает редактировать задачу другим пользователям.

4.3 Меню Вид

4.3.1 Состав меню вид

Меню Вид Мегаполиса имеет следующие разделы:

- Результаты поиска;
- Обзор проекта;
- Объекты;
- Выбор алгоритмов;
- Порядок выполнения;
- Список ошибок.

Ниже описывается более подробно каждый пункт меню Вид.

4.3.2 Результаты поиска

Открывает окно с результатами поиска в Мегаполисе. Окно открывается автоматически при использовании поиска.

4.3.3 Обзор проекта

Пункт меню, открывает окно с набором контроллеров. При помощи контекстного меню, вызываемого нажатием правой кнопкой мыши по контроллеру, создаётся иерархическая структура техпрограммы для выбранного контроллера с переменными, потоками, задачами и подзадачами.

Контекстное меню:

- Объекты – открывает окно с объектами, относящимися к данному контроллеру.
- Создать поток – комплект задач, выполняемых с определённым циклом;
- Создать переменную – открывает окно для создания переменной, в котором необходимо указать набор свойств для новой переменной;
- Устройства ввода-вывода – открывает окно, где необходимо будет создать набор модулей УСО для выбранного контроллера, который будет соответствовать физическому расположению модулей УСО в шкафу контроллера;
- Синхронизация значений – синхронизирует значения между базой данных и контроллером;
- Сделать текущим – позволяет сделать выбранный контроллер текущим. Это позволяет войти в режим обзора, проверить код техпрограммы на наличие ошибок, начать сеанс отладки, приостановить или восстановить выполнение техпрограммы. Иначе эти функции недоступны;
- Вставить – вставляет находящиеся в буфере обмена задачи и подзадачи;
- Диагностика и настройка – открывает окно с полной информацией по контроллеру. Здесь можно посмотреть загрузку ЦП, информацию по загрузке техпрограммы, текущие ошибки, выставленные подмены, межконтроллерный обмен;
- Компиляция и загрузка – окно для компиляции и загрузки (см. п.3.1).

4.3.4 Объекты

Пункт меню открывает окно со списком объектов указанного контроллера.

Все объекты можно отфильтровать по узлу и по типу. Для поиска объектов предусмотрена строка поиска, в которой производится поиск по марке или по имени объекта.

4.3.5 Выбор алгоритма

В ПТК «СУРА» есть встроенная библиотека алгоритмов. Данный пункт меню открывает окно со списком всех алгоритмов. Алгоритмы представлены в виде списка по алфавиту и списка с сортировкой по группам. По умолчанию окно открывается справа в приложении Мегаполиса.

Для того, чтобы добавить алгоритм, необходимо открыть окно задачи, выбрать нужный алгоритм на панели алгоритмов и, зажав левую кнопку мыши, перетащить выбранный алгоритм по полю задачи на нужное место.

4.3.6 Порядок выполнения

Данный пункт открывает окно, в котором представлен порядок выполнения алгоритмов. Список формируется автоматически по мере добавления алгоритмов в поле задач.

В процессе составления техпрограммы может возникнуть необходимость изменить порядок выполнения алгоритмов. Изменить порядок алгоритмов в задаче можно в окне «Порядок алгоритмов», передвинув алгоритм на нужную позицию.

4.3.7 Список ошибок

Пункт меню открывает окно с сообщениями от компилятора техпрограммы. Сообщения делятся на три вида: ошибки, предупреждения, сообщения. Путём нажатия левой кнопкой мыши по названию сообщений можно скрывать или показывать сообщения в окне. Также в окне отображается полоса состояния компиляции с соответствующим описанием.

При нажатии правой кнопкой мыши по одному из сообщений вызывается контекстное меню, которое позволяет выделять сообщения, а также перейти к источнику сообщения или открыть окно, где будет указан путь и возможность перейти к источнику сообщения.

4.4 Свойства задач

Для работы с определенным контроллером нужно в окне Обзора проекта выбрать нужный контроллер.

4.4.1 Список задач

После того, как контроллер выбран, можно создать список задач и подзадач, дополнить этот список новыми задачами или выбрать нужную задачу из готового списка.

Чтобы создать задачу для выбранного контроллера необходимо создать поток для контроллера, а уже в потоке при помощи контекстного меню добавить задачу. Контекстное меню вызывается нажатием правой кнопкой мыши по потоку или в дальнейшем по задаче. При помощи контекстного меню можно выполнить следующие действия:

- создать новую задачу или подзадачу;
- создать переменную для задачи;
- скопировать задачу;
- вставить задачу;
- удалить задачу;
- переместить задачу;
- изменить свойства задачи.

Когда создается новая задача, она помещается в конец списка задач с заданным именем. Аналогичные правила действуют относительно создания подзадач. Разница лишь в том, что в этом случае следует предварительно выбрать родительскую задачу.

Любую задачу можно удалить. Когда задача удаляется, удаляются также все относящиеся к ней подзадачи и все входящие в них алгоблоки.

4.4.2 Поле задачи

Используя панель инструментов, можно менять масштаб изображения текущей задачи. Его можно на несколько ступеней уменьшить, при этом в пределах экрана разместится большее число алгоблоков. Масштаб уменьшенного изображения можно ступенями увеличивать или сразу вернуться к исходному размеру.

Если алгоблок перемещать за пределы видимой части экрана, поле задачи автоматически расширяется и становится больше поля экрана. Поле задачи можно сделать сколь угодно большим (как по вертикали, так и по горизонтали).

Прокручивать поле задачи можно либо обычными средствами операционной системы.

В Мегаполисе имеется функция отката, которая после удаления и перемещения алгоблоков, связей и констант позволяет вернуться к предыдущему, а после отката – к последующему состоянию (два значка с изогнутыми стрелками). Число ступеней отката логически не ограничено (физическое ограничение связано с размером памяти компьютера).

4.4.3 Размещение алгоблоков

Контроллер содержит библиотеку алгоритмов, каждый из которых можно использовать многократно в одной или разных задачах. Алгоритм, помещенный в задачу, называется алгоблоком (алгоритмическим блоком). Таким образом, алгоблок - это работающий алгоритм.

Когда открывается новая задача, первое, что надо сделать, это разместить в поле задачи, требуемые алгоблоки. Для этого выбирается значок «+» контекстного меню поля задач или открывается окно выбор алгоритма

Ряд алгоритмов имеет специальный атрибут, называемый модификатором. Обычно он задает число однотипных элементов, например, число входов сумматора и т.п. Требуемое значение модификатора устанавливается в отдельном поле Модификатор.

Выбранный алгоритм превращается в алгоблок, который первоначально размещается в любом месте поля задач. Захватив алгоблок мышью за верхний заголовок (шапку), его можно переместить в любую другую позицию.

Установленный алгоблок можно удалить или заменить в нем модификатор. Для этого правой клавишей мыши щелкается по шапке алгоблока и из открывающегося контекстного меню выбирается команда соответственно Удалить или Изменить модификатор. При удалении алгоблока все его связи также удаляются. При изменении модификатора связи «отрезанных» каналов удаляются, а оставшихся – сохраняются.

4.4.4 Реквизиты алгоблоков

Алгоблок всегда имеет вид прямоугольника. Его ширина и высота зависит от вида алгоритма, а также от значения модификатора.

Есть алгоритмы, для которых вручную настраивается Тип данных и значение Качества. В первом случае возле шифра алгоритма в <> будет указан тип данных алгоритма, во втором возле порядкового номера будет стоять значок качества.

В верхней части (шапке) алгоблока указываются:

- шифр алгоритма;
- имя алгоритма;
- порядковый номер, определяющий порядок выполнения программы алгоритма;
- значение качества;
- тип данных.

Имя алгоблока присваивается пользователем. Для этого в области шапки нажимается правая клавиша мыши и в открывшемся контекстном меню выбирается строка Имя алгоблока.

По мере добавления алгоблоков в задачу их номера назначаются последовательно – первый добавленный алгоблок получает первый порядковый номер, второй – второй порядковый номер и т.д. Однако затем порядок выполнения алгоблоков можно изменить.

4.4.5 Входы-выходы алгоблока

Ниже шапки размещаются входы-выходы алгоблока, которые имеют обозначения, соответствующие их описанию в библиотеке алгоритмов контроллера.

Помимо обозначения для каждого входа указывается формат, в котором в Мегаполисе представляются значения входных сигналов. Этот формат можно изменять, однако в большинстве случаев его можно оставить тем, который в Мегаполисе принят по умолчанию.

4.4.6 Копирование алгоблоков

В технологической программе контроллера можно копировать выделенный алгоблок и затем вставлять его внутри исходной задачи или в другие задачи контроллеров в рамках проекта. Эта операция осуществляется через буфер обмена.

4.4.7 Свойства входов алгоблоков

Входы алгоблоков имеют следующие общие свойства:

На большинстве входов любого алгоблока можно задать константу либо связать этот вход с каким-либо источником. В первом случае вход называется свободным, во втором – связанным. В отдельных случаях вход нельзя связывать и на нем всегда задается константа. Напротив, некоторые входы предназначены исключительно для связывания и на них нельзя задавать константу.

Конфигурируемый вход любого алгоблока можно соединить с любым выходом любого алгоблока (в том числе со своим собственным выходом). Связывать между собой можно как алгоблоки одной задачи, так и алгоблоки, принадлежащие разным задачам.

С одним входом можно связать только один выход. Это означает, что у одного приемника может быть только один источник. В то же время, к одному выходу можно присоединить любое число разных входов одного или разных алгоблоков.

На законфигурированном входе можно установить инверсию. Для дискретного входа инверсия означает замену логического нуля на единицу, а логической единицы – на ноль. Для аналогового и других непрерывных сигналов инверсия означает изменение знака на противоположный, а для векторного входа – побитную инверсию. На входе, имеющем формат времени, инверсию устанавливать не следует. На свободном входе (с константой) инверсия не задается.

4.4.8 Общий случай установки констант

В исходном состоянии на входах алгоблоков константы равны значениям, указанным в описании библиотеки алгоритмов. На любом свободном (не связанном конфигурацией) входе алгоблока эти константы можно изменять. Совокупность установленных констант определяет параметры настройки алгоблока. Значения констант выводятся слева от полей соответствующих входов алгоблока.

4.4.9 Выбор типа данных

Существует два вида алгоблоков: типизированные и нетипизированные. Для типизированных алгоблоков, по умолчанию, заложен определённый тип данных для всех его переменных, который нельзя изменить, например, ВводАналоговый. Нетипизированные не имеют определенного типа данных, например, Выбор, но для них его можно установить вручную при помощи панели инструментов. Тип данных автоматически подхватывается при проведении связи между типизированным и нетипизированным алгоблоком..

4.5 Конфигурирование алгоблоков

4.5.1 Техника конфигурирования

При конфигурировании, чтобы соединить вход с каким-нибудь выходом, следует подвести мышь к требуемому выходному полю (источнику) - курсор примет вид руки - и затем щелкнуть мышью - рука повернется в другую сторону. Теперь следует щелкнуть мышью на требуемом входном поле (приемнике), - выбранный выход соединится с выбранным входом, а курсор вернется к стандартному изображению. Такой способ, называемый автоматическим, проводит связь по оптимальному маршруту.

Маршрут можно изменить и после того, как связь проведена. Все отрезки линии связи, кроме первого и последнего, можно перемещать с помощью мыши. Для этого к перемещаемому отрезку подводится курсор, - он примет вид двойной стрелки. Теперь, «зацепив» вертикальный

отрезок мышью, его можно переместить влево или вправо, а горизонтальный отрезок - вверх или вниз. Еще одна возможность - подвести курсор к первому или последнему отрезку, - он примет вид перекрестья.

4.5.2 Ссылки

Если связываемые алгоблоки находятся далеко друг от друга, линии связи получаются длинными и вся структура становится плохо читаемой. В таком случае явную связь целесообразно заменить на ссылку. Для этого нужно поместить курсор на любую точку связи, нажать правую клавишу мыши и из контекстного меню выбрать команду Заменить на ссылку, - линия, соединяющая алгоблоки, исчезнет и вместо нее появятся две ссылки: одна рядом с выходом источника, другая – рядом со входом приемника.

Ссылка, «прикрепленная» к источнику, содержит сведения об алгоблоке-приемнике, ссылка, «прикрепленная» к приемнику, содержит аналогичные сведения об алгоблоке-источнике.

Ссылку на входе и выходе алгоблока можно перемещать. Это особенно удобно в том случае, когда с одним выходом связано несколько ссылок на разные приемники, - чтобы эти ссылки не накладывались друг на друга, их следует «расташить».

4.6 Перемещение и перестановка алгоблоков

4.6.1 Возможности перемещения

После того как алгоблок введен в поле задач, его можно перетащить в любое другое место поля задачи (но нельзя переместить в другую задачу). Перемещать можно и законфигурированный алгоблок – подходящие к нему связи будут следовать за алгоблоком.

4.6.2 Наложение алгоблоков и ссылок

При перемещении алгоблоков может произойти их наложение – один алгоблок закроет изображение другого алгоблока. В этом случае перекрытый элемент (алгоблок или ссылка) будет окрашен фиолетовым цветом.

Во всех случаях рекомендуется расташить наложенные элементы.

4.6.3 Перестановка алгоблоков

Размещение алгоблоков в поле задачи никак не определяет порядок их выполнения. Этот порядок вначале целиком определяется последовательностью ввода алгоблоков – вначале выполняется первый введенный алгоблок, затем второй и т.д. Порядковый номер алгоблока (не путать с его адресом) указывается в середине справа шапки алгоблока.

4.7 Комментарии

4.7.1 Виды комментариев

Для того, чтобы в техпрограмме легче было ориентироваться, ее рекомендуется снабжать комментариями. Работая с Мегаполисом, пользователь может использовать три типа комментариев:

- к задаче;
- к алгоблоку;
- к линии связи в виде ссылок для источника и приёмника.

4.7.2 Комментарии к задаче

Комментарии к задаче помещаются в любом свободном месте поля задачи. При наборе текста можно использовать любые символы, имеющиеся на стандартной клавиатуре.

Для того, чтобы ввести комментарий, следует щелкнуть правой клавишей мыши в нужном месте свободного поля задачи и в открывшемся диалоговом окне выбрать строку Добавить комментарий. После этого в выбранном месте поля задачи появится окно Комментарий, в которое можно ввести текст. Когда текст введен, следует нажать кнопку ок – после чего текст помещается в поле задачи. Затем этот текст с помощью мыши можно перетащить в любое другое место.

4.7.3 Другие виды комментариев

Аналогично комментариям к задаче можно вводить комментарии к каждому алгоблоку и к линии связи в виде ссылок между алгоблоками, причем в последнем случае можно вводить два разных (или одинаковых) комментария к источнику и приемнику сигнала. Введенный текст можно переместить в другое место, для текста можно выбрать требуемый шрифт. Особенность этих комментариев заключается в том, что текст «привязывается» к месту расположения алгоблока – при перемещении алгоблока соответствующим образом перемещаются как комментарии к данному алгоблоку, так и комментарии к ссылкам, подходящим к данному алгоблоку.

4.8 Привязка

4.8.1 Назначение привязки

Термином привязка обозначаются процедуры, связывающие алгоблок или его отдельные элементы с объектами базы данных. Техника привязки зависит от свойств алгоблока – является ли он объектным или обычным, одноканальным или многоканальным.

В библиотеке имеется группа алгоритмов, называемых объектными. Эти алгоритмы формируют обобщенную информацию о свойствах реальных или виртуальных объектов (датчиков, механизмов, регуляторов и т.п.). Чтобы информацию о таких объектах передавать оперативным станциям (операторской станции, архиву и т.д.), соответствующие алгоблоки необходимо привязать к тем или иным объектам в базе данных.

В качестве объекта может выступать отдельный аналоговый, дискретный, числовой и временной сигнал, вычисленный в контроллере. В таком случае к объектам базы данных привязывается отдельный выход или вход алгоблока, отражающий требуемый сигнал, а если сигнал векторный, то и отдельный бит, который привязывается как дискретный сигнал.

4.8.2 Отображение привязки

В Мегаполисе предусмотрена специальная метка, которая указывает на привязанные элементы. Рядом с меткой добавляется название марки.

4.9 Права доступа и многопользовательская работа с Мегаполисом

4.9.1 Права доступа

Права доступа определяют, что может и что не может делать конкретный пользователь. Права доступа к Мегаполису прописываются в программе. В ней для каждого пользователя задаются индивидуальные права на работу со следующими процедурами Мегаполиса:

- редактирование техпрограммы;
- редактирование дерева задач;
- компиляция техпрограммы;
- загрузка техпрограммы.

Если пользователь не имеет прав, он не сможет выполнить соответствующую процедуру, однако отсутствие прав не препятствует просмотру задач.

4.9.2 Многопользовательская работа

В разработке одного проекта может участвовать несколько пользователей. Чтобы сохранить целостность БД, используется механизм блокировок, которые в большинстве случаев ставятся автоматически. В Мегаполисе реализуются следующие возможности многопользовательской работы:

Несколько пользователей могут одновременно редактировать разные задачи как одного и того же, так и разных контроллеров.

Несколько пользователей могут просматривать одну и ту же задачу, но только один (тот, который начал редактирование) может продолжать ее редактировать. Как только в задаче что-то изменилось, она блокируется и пока не будет закрыта, никто другой ничего не может в ней изменить.

На время редактирования дерева задач в данном контроллере в режиме Обзор блокируется возможность такого редактирования для других пользователей. Эта возможность появляется по окончании редактирования дерева задач.

Если один пользователь запустил компиляцию, загрузку или блокирующий обзор контроллера, то пока не завершились эти процессы, другие пользователи не могут запустить эти операции в том же контроллере и не могут редактировать ни одну из задач данного контроллера.

5 Отладка техпрограмм

Для отладки созданной средствами Мегаполиса техпрограммы в реальном времени при ее работе в контроллере, программу необходимо скомпилировать, загрузить в контроллер и включить режим работы Обзор, при котором Мегаполис периодически запрашивает состояния алгоблоков программы и выводит их значения на экран.

5.1 Компиляция и загрузка техпрограммы

Компиляция техпрограммы – это процедура, которая исходную графическую форму техпрограммы, представленную в виде задач, алгоблоков, связей и констант, преобразует в единый файл, «пригодный» для загрузки в контроллер. Этот файл сохраняется в базе данных и в любое время может быть загружен в контроллер.

Для того чтобы проверить техпрограмму на наличие ошибок без компиляции и загрузки, в окне необходимо нажать кнопку проверить.

Перед компиляцией необходимо произвести настройку компилятора.

После настройки компилятора можно приступать к компиляции. Установив галочку «Собрать технокод» и нажав кнопку выполнить Мегаполис скомпилирует техпрограмму с учётом установленных настроек компилятора. Чтобы загрузить подготовленную технологическую программу в контроллер, рабочая станция с Мегаполисом и контроллер должны находиться в общей сети. Технологическая программа загружается в контроллер целиком, при этом загружаются все подготовленные задачи, не зависимо от того, включены они в Мегаполисе или отключены. Чтобы загрузить программу в контроллер, в окне компиляции и загрузки устанавливается галочка «Загрузить технокод» и нажимается кнопка Выполнить.

«Безопасная загрузка» позволяет загружать техпрограмму в оперативном режиме. Если установлена опция «Значения из технокода», то значения на входах алгоблоков и значения ручных переменных будут взяты из технокода, развернутого на сервере. Иначе – они будут браться из сохраненного кэша данных.

5.2 Безопасная загрузка

В ПТК «СУРА» реализована возможность конфигурирования проекта и расширения системы без останова технологического процесса.

Уставки, режимы, шкалы и прочие настроечные параметры технологической программы контроллера могут быть изменены онлайн.

Более глобальное изменение технологической программы контроллера (например, добавление новых алгоритмов, изменения в составе модулей УСО, конфигурация каналов УСО и сигналов, привязанных к ним) требует перезагрузки контроллера. Для выполнения данной процедуры без останова технологического процесса реализован механизм «Безопасной загрузки» контроллера.

«Безопасная загрузка» - опция, которая применяется для безударной загрузки новой технологической программы в контроллер в процессе управления технологическим процессом. При использовании этой опции новая технологическая программа загружается в пассивную компоненту дублированного контроллера, запускается и проверяется. В случае, если новая технологическая программа прошла все проверки, автоматически выполняется переключение активности на пассивную компоненту дублированного контроллера, последующая пересылка новой технологической программы и синхронизация данных в бывшую активную компоненту (ставшую после переключения активности пассивной). Таким образом обеспечивается безударная загрузка новой технологической программы в контроллер (время переключения активности составляет несколько миллисекунд). При этом на все время загрузки (в зависимости от размера технологической программы. Обычно порядка 40 секунд) контроллер фактически работает в одиночном режиме. Данный вариант загрузки технологической программы возможен, если новая технологическая программа, загруженная в пассивную компоненту дублированного контроллера, по результатам диагностики признается не хуже (формирует такой же или меньший интегральный уровень отказа), чем текущая технологическая программа. В противном случае происходит «откат» всех изменений и возвращение контроллера на работу со старой технологической

программой. Если есть необходимость провести загрузку, игнорируя диагностику уровней отказов старой и новой технологической программы, существует дополнительный вид «Безопасной загрузки», которая называется «Принудительная загрузка» (см. ниже).

При расширении системы и добавлении модулей УСО бывают ситуации, когда технологическое оборудование объекта (датчики, исполнительные механизмы) ещё не подключено или должным образом не настроено, а связь данного оборудования с модулями УСО уже реализована. В таком случае, при добавлении в проект модулей УСО и последующей загрузке технологической программы, система диагностики контроллера выявит на входе модулей нештатную ситуацию, повысит уровень отказа контроллера, и стандартная безопасная загрузка не будет выполнена. Для таких случаев рекомендуется применять «Принудительную загрузку» (разновидность «Безопасной загрузки»), если изменения необходимо внести на работающем технологическом оборудовании.

В случае использования режима «Принудительная загрузка» переключение активности на пассивную компоненту дублированного контроллера происходит даже, если новая технологическая программа по результатам внутренней диагностики контроллера признана хуже (формирует больший интегральный уровень отказа), чем текущая технологическая программа. Такой вариант допустим, например, в вышеуказанном случае при добавлении модулей ввода-вывода на работающем технологическом оборудовании.

Опция «Безопасная загрузка» применима для контроллера, работающего в режиме дублирования, и недоступна для одиночного исполнения контроллера.

5.3 Обзор

5.3.1 Возможности обзора

Обзор позволяет просмотреть непосредственно в Мегаполисе значения «живых» сигналов, формируемых контроллером. Как и при загрузке, при обзоре рабочая станция с Мегаполисом должна находиться в общей сети с контроллером.

Чтобы перевести в режим обзора или вывести из обзора всю задачу, нужно из главного меню выбрать команду соответственно «Войти в режим обзора» или «Выйти из режима обзора». Обзор задачи означает обзор всех входов и выходов всех алгоблоков выбранной задачи. Отмена обзора задачи означает отмену обзора всех входов-выходов всех алгоблоков данной задачи.

В режиме обзора рядом с выбранным полем входа (выхода) появится обозначенное зеленым цветом «живое» значение сигнала, полученное непосредственно из контроллера.

6 Рекомендации по подготовке техпрограмм

6.1 Ввод/вывод информации через модули УСО

Для физического ввода-вывода сигналов в контроллере используются специальные модули – устройства связи с объектом (УСО).

Ограничения по модулям УСО:

- число блоков УСО в одном контроллере – не более 25 (с учетом расширения);
- число модулей УСО в пределах одного блока – не более 15 (для АЦП не более 12).

6.2 Межконтроллерный обмен

Осуществляется путём проведения связи из одного контроллера в другой.

Ограничения по межконтроллерному обмену:

- число подключенных к данному контроллеру по сети других контроллеров – не более 10 одиночных или не более 5 дублированных.

6.3 Объектные алгоритмы

6.3.1 Информация об объектах

В СУРЕ объект – это совокупность параметров, характеризующих некоторый физический или виртуальный элемент АСУ ТП как единое целое. Эти параметры могут быть представлены на экране операторской станции и/или записаны в архив. Объект обязательно должен иметь марку, уникальную (не повторяющуюся) в пределах одной интегрированной АСУ ТП.

В СУРЕ существует классификация объектов по так называемым оперативным типам. Имеется предопределенный перечень оперативных типов объектов (Ввод Аналоговый, Регулятор, Задвижка и т.д.), - всего несколько десятков типов.

6.3.2 Общие свойства объектных алгоритмов

Помимо своих «прямых» функций, связанных с работой технологической программы, объектные алгоритмы выполняют ряд информационных функций:

Они формируют совокупную информацию о значениях параметров и текущем состоянии объекта (открыто/закрыто, автоматический/ручной, достоверность и т.д.). Эти сведения передаются по запросам операторской станции и отображаются на мнемосхемах или оперативных окнах.

Они направляют в архив информацию о появлении или исчезновении штатных событий (изменении состояний, режимов и т.п.) и появлении или исчезновении нештатных событий, т.е. ошибок.

Они направляют рабочим станциям информацию о текущих нештатных событиях, т.е. ошибках (недостоверности параметров, невыполнении условий и т.п.). Эту информацию может принять любая станция, в которой предусмотрена процедура представления ошибок (к таким станциям, в частности, относится операторская и событийная станции).

Специально подготавливать в контроллере оперативную информацию о таких объектах не требуется. После того, как алгоритм с помощью Мегаполиса привязан к марке объекта, вся необходимая информация готова для передачи станциям (это утверждение не касается периодической записи в архив).

6.4 Вычислительные алгоритмы

6.4.1 Состав вычислительных алгоритмов

Библиотека контроллеров содержит целый ряд алгоритмов, с помощью которых можно выполнять аналоговые и дискретные вычисления. Для аналоговых статических и динамических вычислений используются алгоритмы суммирования, масштабирования, извлечения корня, интегрирования и т.п. Дискретные вычисления выполняются с помощью алгоритмов логического И, ИЛИ, различного рода триггеров и т.д.

6.5 Архивирование сигналов

6.5.1 Варианты архивирования

В СУРЕ реализованы 2 метода архивирования сигналов:

- периодическое;
- апертурное.

Периодическое архивирование сигналов выполняется регулярно с заданным периодом времени. При апертурном архивировании сигналы записываются в архив только при их изменении на величину больше апертур.

6.5.2 Периодическое архивирование сигналов

При периодическом архивировании значения сигналов записываются в архив с заданным периодом не зависимо от того, насколько изменился сигнал в течение этого периода.

По умолчанию периодическое архивирование отключено. Для того, чтобы выбранный сигнал записывался в архив, необходимо принудительно изменить значение атрибута **Период**.

Архивироваться могут сигналы, сформированные любым как объектным, так и неobjектным алгоритмом. При этом следует иметь ввиду, что периодически архивировать имеет смысл в основном аналоговые сигналы. Дискретные признаки объектных алгоритмов «автоматически» архивируются в виде событий, а для дискретных сигналов неobjектных алгоритмов можно такие события запрограммировать.

6.5.3 Архивирование аналоговых сигналов по апертуре

При апертурном архивировании аналоговый сигнал записывается в архив, только если он изменился на определенную величину (апертур) относительно значения, ранее записанного в архив.

Архивирование по апертуре уменьшает объем архива и снижает нагрузку на информационную сеть. В то же время, если архивируемые по апертуре сигналы редко выходят за установленные апертурные значения, в архиве будет слишком мало записей и их неудобно будет анализировать. Поэтому апертурный метод рекомендуется сочетать с периодической записью в архив тех же сигналов, при этом период записи можно выбрать достаточно большим (однако, не настолько, чтобы потерять наглядность представления трендов).

6.5.4 Объектное архивирование

Большинство объектных алгоритмов формируют события, которые «автоматически» (без специального программирования) направляются в архив. Например, в архив автоматически направляются события Задвижка открылась (закрылась), Двигатель включился (отключился) и т.д. Аналогично в архив записываются и все ошибки, фиксируемые объектными алгоритмами. Подобные записи в архив инициирует сам контроллер.

Лист регистрации изменений

[illegible]