

# Универсальная среда разработки Agava

---

Универсальная среда разработки предназначена для создания и редактирования программного обеспечения для панелей оператора АГАВА ПО, многоканальных программируемых реле АГАВА МПР-60 и систем диспетчеризации на базе SCADA-системы AgavaSCADA.



## Содержание

---

### Введение

Версии программного обеспечения

Дополнительные статьи и документы

### Разработка проекта

Создание нового проекта

Редактирование существующего проекта

Структура и состав проекта

Разработка проекта

Взаимодействие с внешними устройствами

Визуализация данных

Пользовательские алгоритмы

Редактирование свойств узлов

Редактирование свойств узлов с помощью окна «Свойства»

Редактирование свойств группы узлов

Групповое редактирование свойств узлов

Функция групповой замены источников в узлах

Редактирование графического представления узла

Связывание источников и виджетов

Редактирование многослойных композиций

Использование якорей

Сохранение проекта

### Проверка и отладка проекта

Эмуляция проекта

Мониторинг сигналов

Мониторинг событий

Использование мониторинга

Фильтрация событий

Отладка в задаче ПЛК

Управление отладкой

Режим симулятора

Онлайн отладка

Компиляция проекта

Проверка проекта

Установка пароля на файл конфигурации

Загрузка проекта в устройство

Загрузка из среды разработки

Загрузка через USB-Flash (для ПО-40 и ПО-50)

Загрузка через браузер (для МПР-60)

Отладка проекта

Локальная отладка

Удаленная отладка

## 1 Введение

---

Универсальная среда разработки Agava имеет удобный и эргономичный пользовательский интерфейс со всеми инструментами, необходимыми для создания эффективных проектов-приложений, предназначенных для исполнения на ПК, панелях оператора или персональных компьютерах.

Среда разработки совмещает редакторы текста, редакторы визуальных представлений объектов, компоновщики структуры, средства связывания объектов и других частей проекта, анализаторы проекта в одном приложении и дает пользователю возможность ведения эффективной разработки проектов-приложений от начала и до завершения в одной программе, без использования разнородных приложений-утилит и необходимости переключения между ними.

### 1.1 Версии программного обеспечения



При использовании среды разработки совместно с панелями оператора АГАВА ПО или многоканальными программируемыми реле АГАВА МПР-60 необходимо использовать соответствующую версию среды разработки

Для определения подходящей версии среды разработки необходимо определить версию среды исполнения, установленной в панели оператора или реле. Тип и версия среды исполнения указаны в системной утилите прибора. Например, "AgavaPLC 1.1", "AgavaPLC 1.2" и т.д. Для использования со средой исполнения AgavaPLC 1.1 необходимо использовать среду разработки версии 1.1, для среды исполнения AgavaPLC 1.2 необходима среда разработки версии 1.2 и так далее.

### 1.2 Дополнительные статьи и документы

- Описание узлов AgavaSCADA/AgavaPLC.
- Свойства узлов в AgavaSCADA/AgavaPLC.
- Транспорты AgavaSCADA/AgavaPLC.
- Описание языка C++ в AgavaSCADA/AgavaPLC..
- Объектная модель AgavaSCADA/AgavaPLC.
- Описание базовых классов AgavaSCADA/AgavaPLC.
- Коды ошибок и предупреждений AgavaSCADA/AgavaPLC.
- Рекомендации по разработке приложений AgavaSCADA/AgavaPLC.

Полный перечень всех документов, связанных с разработкой приложений на AgavaSCADA/AgavaPLC, приведен на странице [категории AgavaSCADA/AgavaPLC.](#)

## 2 Разработка проекта

---

### 2.1 Создание нового проекта

Создание нового проекта в среде разработки осуществляется с помощью команды «Создать проект...» в меню «Файл». После вызова данной команды откроется диалоговое окно создания нового проекта (см. Рисунок 1).

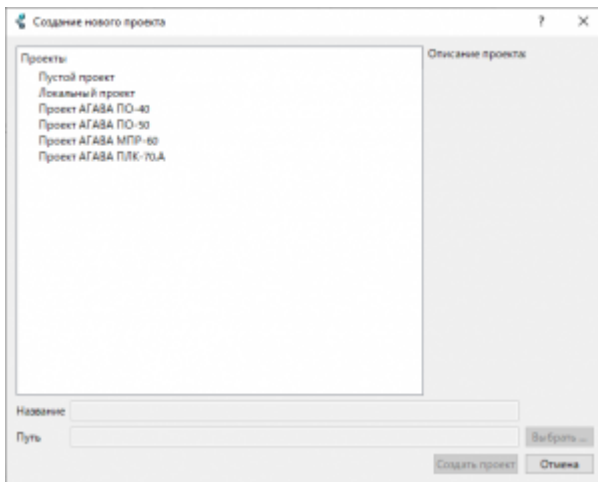


Рисунок 1 - Окно "Создание нового проекта"

Для создания проекта необходимо в списке типов проектов выбрать необходимый, ввести имя проекта и путь для его сохранения, после чего нажать кнопку «Создать проект».

Доступны несколько типов проектов:

- «Пустой проект» – пустой проект, не содержащий никаких узлов. Данный тип можно использовать для создания произвольного проекта "с нуля".
- «Локальный проект» – предварительно настроенный проект, содержащий необходимые подсистемы и предназначенный для реализации простой системы для одного устройства, при выборе данного типа пользователь имеет возможность выбрать нужные ему подсистемы и получить полностью рабочий проект в

минимальной конфигурации.

- «Проект АГАВА ПО-40» - проект, предназначенный для исполнения на панели оператора АГАВА ПО-40.
- «Проект АГАВА ПО-50» - проект, предназначенный для исполнения на панели оператора АГАВА ПО-50.
- «Проект АГАВА МПР-60» - проект, предназначенный для исполнения на контроллере АГАВА МПР-60.
- «Проект АГАВА ПЛК-70.А» - проект, предназначенный для исполнения на контроллере АГАВА ПЛК-70.А.

## 2.2 Редактирование существующего проекта

Редактирование существующего проекта начинается с открытия файла проекта путем вызова команды «Открыть...» в меню «Файл». После загрузки файла в основном окне среды разработки будут отображены свойства и параметры проекта (см. Рисунок 2).

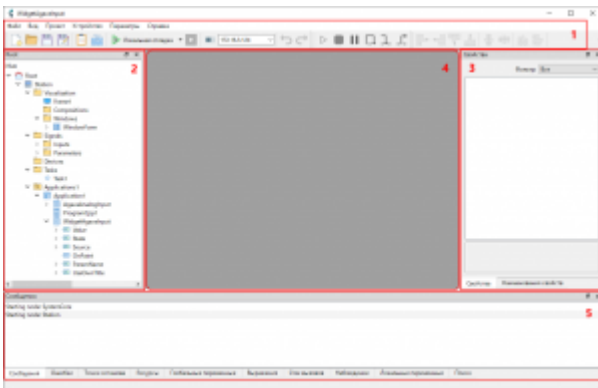


Рисунок 2 – Основное окно среды разработки AgavaPLC с загруженным проектом

По умолчанию основное окно среды разработки содержит перечисленные ниже части:

1. Строка заголовка приложения, панель меню и панель инструментов.
2. Окно дерева проекта. Предоставляет доступ ко всей структуре проекта (см. раздел Узлы проекта).
3. Окно свойств (см. раздел #Редактирование свойств узлов ).
4. Область для отображения содержимого, например графического представления окна, текста программы или задачи ПЛК.
5. Область вспомогательных окон:
  1. Окно "Сообщения".
  2. Окно "Ошибки".
  3. Окно "Точки останова".
  4. Окно "Ресурсы".
  5. Окно "Глобальные переменные".
  6. Окно "Выражения".
  7. Окно "Стек вызовов".

8. Окно "Наблюдение".
9. Окно "Локальные переменные".
10. Окно "Поиск".

После открытия проекта можно приступить к его редактированию.

По завершению всех операций по редактированию необходимо выполнить проверку проекта на корректность с помощью команды «Проверить проект», расположенной в меню «Проект» (см. 6.1).

## 2.3 Структура и состав проекта

Типичный проект содержит следующие части:

- станция – важнейший объект, управляющий работой собственных подсистем и осуществляющий обмен данными с другими станциями;
- транспортная подсистема, выполняющая сбор данных с УСО;
- подсистема визуализации, обеспечивающая отображение информации в различных видах.

Проект также может содержать следующие части:

- подсистема архивирования, сохраняющая данные в БД для последующего извлечения и анализа;
- подсистема проверки прав пользователей, предназначенная для контроля доступа пользователя к объектам в режиме исполнения;
- подсистема тревог, предназначенная для оповещения пользователя об обнаруженных нарушениях;
- подсистема журналирования, выполняющая архивирование зарегистрированных событий в БД для последующего извлечения и анализа.

Проект имеет древовидную структуру, вследствие чего все узлы связаны друг с другом. Узлы нижних уровней подчиняются узлам верхних уровней. К примеру, станции подчинены все подсистемы, а каждой подсистеме подчинены все узлы, которыми она управляет.

Описание всех частей проекта и их узлов приведено в разделе 4.

## 2.4 Разработка проекта

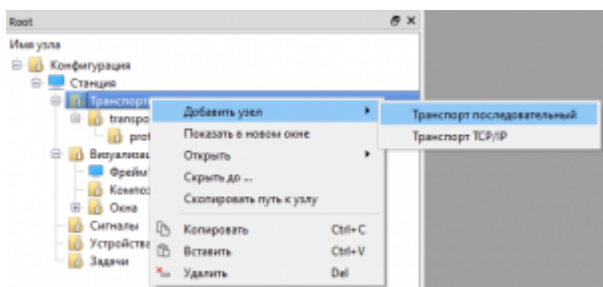


Рисунок 3 – Добавление узла в проект  
AgavaPLC

Перед началом разработки проекта рекомендуется проанализировать поставленную задачи и определить подсистемы, необходимые для её реализации. Взаимодействие с подсистемами осуществляется через дерево проекта (см. Рисунок 2), в которое добавляются нужные узлы. Для добавления нового узла, необходимо нажать правой кнопкой мыши на узел верхнего уровня, которому он подчинён, и выбрать в контекстном меню команду «Добавить узел».

Основные части типовой программы и способы их реализации приведены ниже.

### 2.4.1 Взаимодействие с внешними устройствами

Реализуется через подсистему «Транспорты» в окне дерева проекта. Необходимо добавить в нее нужные протоколы, в зависимости от имеющейся конфигурации контроллера.

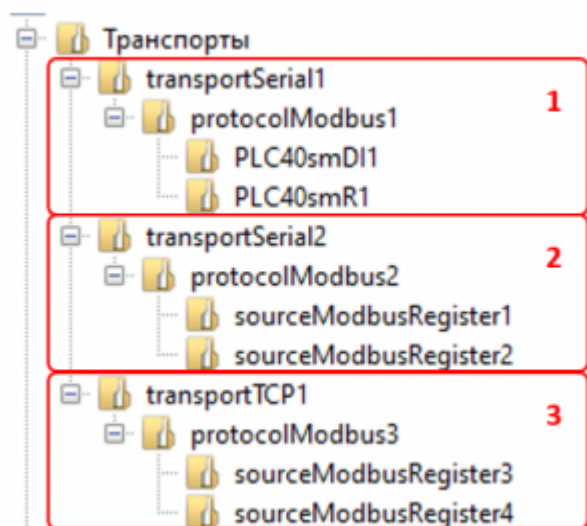


Рисунок 4 - Транспорты в дереве проекта  
AgavaPLC

1. Транспорт для обмена с submodule контроллера (для контроллеров с корзиной submodule). В качестве порта транспорта необходимо указать `"/dev/ttyS0"`. В проекте для МПР-60 присутствует по-умолчанию.
2. Транспорт последовательный (для управления по RS-232, RS-485 и т.д.).
3. Транспорт TCP/IP (для управления по Ethernet).

К транспортам присоединяются либо регистры Modbus, либо submodule. Для регистра задается адрес устройства, номер регистра, функция чтения/записи и другие параметры, при необходимости. Для submodule указывается slot, в котором он установлен, и тип аналоговых входов/выходов.

### 2.4.2 Визуализация данных

Визуализация данных технологических процессов реализуется с помощью элементов

подсистемы «Визуализация».

Система визуализации недоступна в проектах для контроллеров МПР.

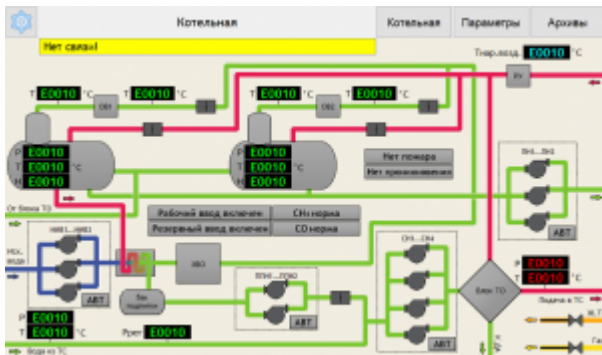


Рисунок 5 - Пример визуализации

Визуализация содержит одно или несколько окон, описанных ниже.

1. Форма, которая может использоваться как меню или панель инструментов. Также применяется в небольших проектах, в которых нет необходимости изменения выводимого набора виджетов.
2. Окно с композицией, которое позволяет переключаться между различными экранами (например мнемосхемой, настройками, архивированием и т.д). Каждый из этих экранов реализуется на отдельной композиции.

Процесс создания визуализации начинается с добавления нужных окон в группу «Окна» в дереве проекта. В том случае, если используется окно с композицией, необходимо задать ему отображаемую композицию по умолчанию, перед этим добавив ее в группу «Композиции». Созданные окна нужно добавить к одному фрейму, указав их в соответствующем свойстве фрейма.

Редактирование содержимого окна или композиции осуществляется в его графическом представлении (см. раздел [#Редактирование графического представления узла](#)).

Более подробное описание подсистемы визуализации приведено в разделе [#Подсистема визуализации](#).

### 2.4.3 Пользовательские алгоритмы

Реализовать нужные алгоритмы работы программы можно с помощью различных инструментов, которые выбираются в зависимости от сложности проектов и навыков программиста:

1. Операции в группе «Сигналы» – отдельные логические операции. В основном используются для небольших проектов, не требующих сложных методов управления;
2. Задачи ПЛК – графическое программирование с помощью связывания между собой различных функциональных блоков. Подходит для больших проектов. Реализуется в узле «Задача ПЛК» группы «Задачи». Подробное описание приведено в разделе [#Задача ПЛК](#).

3. Скрипты С++ – текстовое программирование на языке высокого уровня. Подходит для больших проектов. Реализуется с помощью операций «Скрипт С++», расположенных в группе «Сигналы» или в задаче ПЛК. Описание реализации языка С++ приведено на странице [Описание языка С++ в AgavaPLC](#).

## 2.5 Редактирование свойств узлов

Среда разработки обладает развитыми возможностями по редактированию свойств узлов. Описание доступных типов свойств приведено в статье [Свойства узлов в AgavaPLC/AgavaSCADA](#).

### 2.5.1 Редактирование свойств узлов с помощью окна «Свойства»

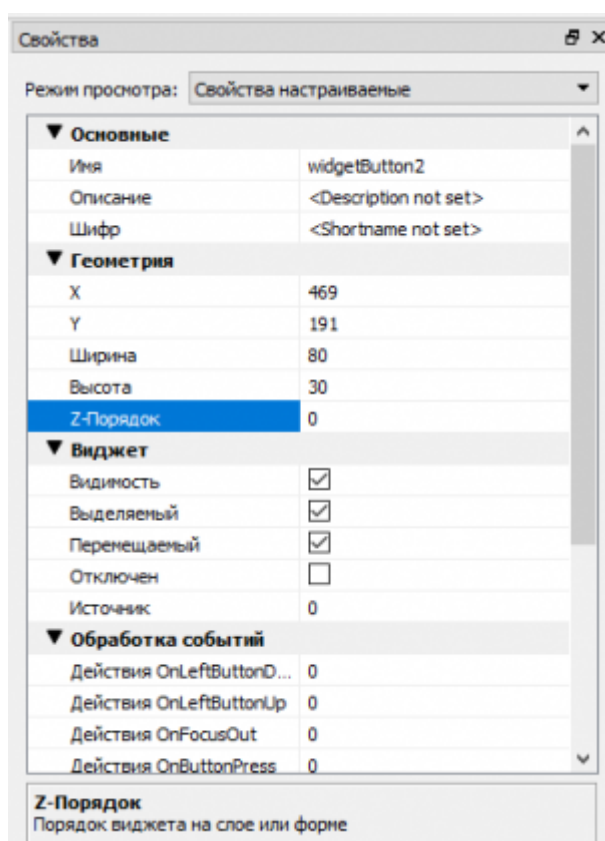


Рисунок 6 – Окно «Свойства» виджета «Кнопка»

Редактирование свойств всех узлов осуществляется с помощью окна «Свойства».

В окне «Свойства» в виде таблицы отображаются все доступные свойства узла, объединенные в группы. В левой колонке отображены наименования свойств, в правой отображены их значения. Свойства, недоступные для изменения, имеют наименование серого цвета. В нижней части окна расположена область подсказки, содержащая пояснение к выделенному свойству.

#### 2.5.1.1 Редактирование свойств группы узлов

В случае, когда необходимо изменить свойства у нескольких узлов, можно воспользоваться возможностью группового редактирования свойств. Для этого нужно

выделить несколько узлов в дереве конфигурации либо на графическом представлении с помощью левой кнопки мыши и нажатой кнопки «Ctrl». При этом в окне свойств будут отображены только те свойства, что имеются у всех выделенных узлов. Далее можно изменить нужное свойство, при этом после завершения редактирования измененное значение будет установлено во всех выбранных узлах.

### 2.5.2 Групповое редактирование свойств узлов

При необходимости редактирования разных свойств у большого количества узлов можно воспользоваться специальным окном «Список узлов».

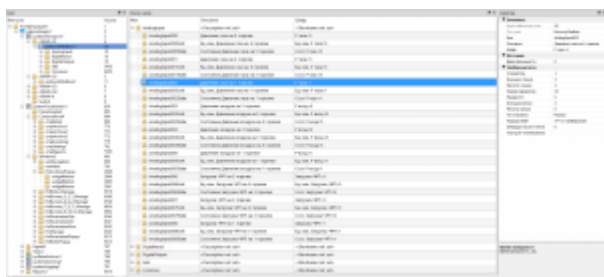


Рисунок 7 - Окно "Список узлов"

Открытие окна выполняется с помощью вызова команды «Открыть список узлов» в контекстном меню дерева конфигурации. Открытое окно содержит все узлы, вложенные в выбранные при вызове команды.

Окно позволяет редактировать свойства непосредственно в ячейках таблицы, без переключения на окно «Свойства». Групповое редактирование свойств узлов через окно «Свойства» также доступно.

Щелчок правой кнопкой мыши на списке вызывает контекстное меню, содержащее команды:

- «Свойства»;
- «Экспорт в \*.txt»;
- «Импорт из \*.txt»;
- «Удалить узел».

Команда «Свойства» открывает диалоговое окно выбора отображаемых свойств.

Команды «Экспорт», «Импорт» выполняют экспорт списка свойств узлов в текстовый файл и импорт соответственно.

Команда «Удалить узел» производит удаление выделенного узла (узлов).

### 2.5.3 Функция групповой замены источников в узлах

При разработке сложных проектов может возникнуть необходимость заменить источники сразу в большом количестве узлов. Для этой цели используется функция групповой замены источников. Необходимо перейти в редактирование свойств типа «Список источников», которое производится в соответствующем окне (см. Рисунок 8).

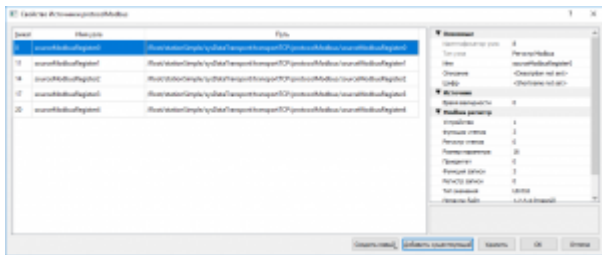


Рисунок 8 - Окно редактирования свойств типа «Список источников»

Затем необходимо использовать функцию групповой замены (см. Рисунок 9), выделив нужные узлы, нажав на них правой кнопкой мыши и выбрав «Групповая замена».

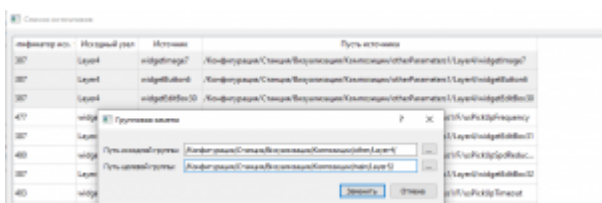


Рисунок 9 - Окно функции «Групповая замена»

## 2.6 Редактирование графического представления узла

Редактирование содержимого окон, слоев, форм, композиций и других элементов, предназначенных для отображения на экране, осуществляется в окне графического представления. Открытие окна возможно с помощью команды «Открыть графическое представление» в контекстном меню дерева конфигурации или двойного нажатия левой кнопки мыши для узлов следующих типов:

- фрейм;
- форма;
- окно с композицией;
- слой.

Открытое графическое представление выбранного узла может содержать все уже размещенные на нем элементы визуализации и (или) другие узлы и позволяет добавить новые.

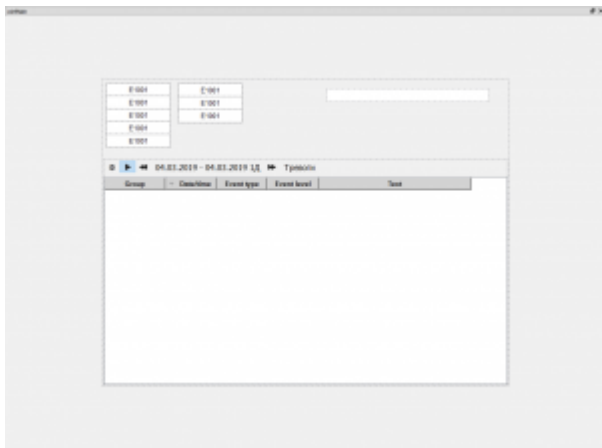


Рисунок 10 – Открытое графическое представление окна «winMain»

На рисунке 10 представлено графическое представление окна «winMain», обведенное пунктирной рамкой серого цвета, обозначающей внешнюю границу открытого узла. Тип рамки пользователь может выбрать по своему усмотрению, выбор производится в окне настроек среды разработки, доступном через команду «Настройки» в меню «Параметры».

Щелчок правой кнопкой мыши на пустом месте в открытом графическом представлении узла вызывает контекстное меню, содержащее команды удаления и вставки виджетов из буфера обмена, а также команды добавления новых виджетов доступных типов.

Добавление новых виджетов также может осуществляться путем их перетаскивания в рабочую область из панели виджетов.

Графическое представление позволяет перемещать выделенные виджеты кнопками перемещения курсора на клавиатуре, а также изменять их размеры кнопками перемещения курсора при нажатой кнопке Shift.

### 2.6.1 Связывание источников и виджетов

Связывание виджетов и других узлов с источниками данных необходимо для того, чтобы виджеты или другие узлы могли отображать или обрабатывать значения, хранящиеся в источниках.

Для связи виджетов с источниками данных можно воспользоваться обычным способом связывания через окно «Свойства». К примеру, связывание размещенного на форме виджета «Текстовая метка» с регистром Modbus осуществляется добавлением регистра Modbus в свойство виджета «Источник».

Вместе с обычным способом связывания существует и другой, более быстрый способ. Для быстрого связывания регистра с виджетом достаточно нажать на регистр в дереве конфигурации левой кнопкой мыши и перенести его на виджет в открытом графическом представлении.

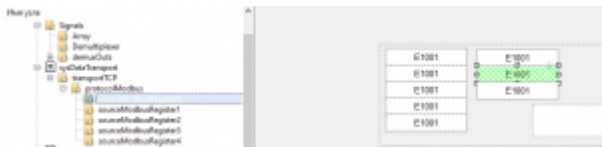


Рисунок 11 – Связывание узла «sourceModbusRegister0» и виджета

При помещении курсора над виджетом последний будет заштрихован сеткой зеленого или красного цвета, а курсор изменит свой вид на стрелку с квадратиком (см. рисунок 11).

Зеленая штриховка виджета означает отсутствие связи выделенного виджета с источником данных, в этом случае связывание будет произведено. Красная штриховка говорит об ошибках связывания виджета и сигнала/источника.

## 2.6.2 Редактирование многослойных композиций

Редактирование многослойных композиций принципиально не отличается от редактирования форм, кроме одного нюанса. Так как композиция может состоять из нескольких слоев, и виджеты на слоях могут располагаться один над другим, для использования наиболее удобного способа редактирования такой композиции можно воспользоваться возможностью открытия каждого слоя в отдельном окне.

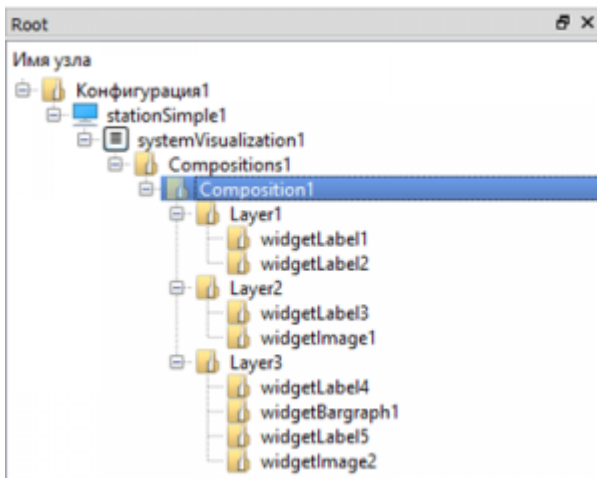


Рисунок 12 – Пример структуры композиции «Composition1», состоящей из трех слоев

Для открытия графического представления слоя нужно воспользоваться командой «Открыть графическое представление» аналогично открытию графического представления формы или композиции.

Открыв одновременно композицию и слой, при редактировании слоя можно видеть, что содержимое композиции также изменяется и наоборот, при редактировании композиции изображение слоя будет изменяться.

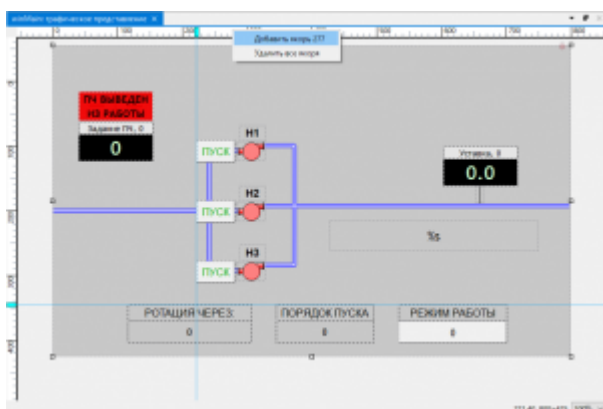
Слой обладает важным свойством «Прозрачность для действий». Данное свойство позволяет регулировать доступность виджетов, расположенных на нижележащих слоях, для действий пользователя. Например, если у верхнего слоя данное свойство выключено, то виджеты, расположенные на всех нижележащих слоях, будут недоступны для взаимодействия. Если же у верхнего слоя свойство включено, то

виджеты нижележащего слоя будут доступны.

### 2.6.3 Использование якорей

Данная возможность доступна с версии AS 1.6.32+.

При работе с графическими представлениями окон и композиций, содержащих большое количество виджетов, могут быть полезны якоря - создаваемые пользователем метки на размерных линейках.



AS 1.6.32+: Графическое представление формы с установленными якорями

Якоря позволяют контролировать положение групп виджетов благодаря связанным с ними линиям привязки. Линии привязки, позволяют быстрее выстраивать группы виджетов благодаря "прилипанию".

Создание якоря выполняется следующими способами:

- Двойным кликом левой кнопки мыши по размерной линейке.
- Кликом правой кнопки мыши на размерной линейке и выбором команды "Добавить якорь".

Удаление якоря выполняется следующими способами:

- Двойным кликом левой кнопки мыши по якорю на размерной линейке.
- Кликом правой кнопки мыши на размерной линейке и выбором команды "Удалить якорь".

Созданные якоря можно перемещать по размерной линейке с помощью мыши.

## 2.7 Сохранение проекта

Для сохранения проекта на диск вызовите команду «Файл->Сохранить проект» или «Файл->Сохранить проект как...», если необходимо сделать копию открытого проекта и продолжить работу в ней после сохранения.

## 3 Проверка и отладка проекта

---

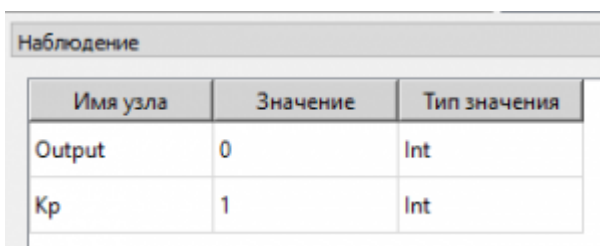
### 3.1 Эмуляция проекта

Проверить работоспособность проекта можно, не загружая его в контроллер. Для этого существует возможность запустить среду исполнения прямо на компьютере.

Для этого необходимо:

1. Установить текущий проект как проект по умолчанию. Для этого необходимо вызвать команду "Проект->Установить проект по умолчанию..." и выбрать ранее сохраненный файл конфигурации.
2. Запустить среду исполнения. Для этого нужно вызвать команду меню "Проект->Запуск среды исполнения". После этого будет выполнен запуск в новом окне среды исполнения.

### 3.2 Мониторинг сигналов



Имя узла	Значение	Тип значения
Output	0	Int
Кр	1	Int

ссылка=Файл:AgavaPLC-Окно\_%"Наблюдени  
e%"e%.png

Функция мониторинга сигналов позволяет наблюдать значения, хранящиеся в узлах проекта при исполнении проекта на целевом устройстве.

Для запуска данной функции необходимо:

1. Запустить среду исполнения.
2. Вызвать команду меню «Устройство → Подключение/отключение», в появившемся окне выбрать IP-адрес и порт устройства, значения узлов которого необходимо мониторить. При необходимости просмотра значений узлов и сигналов в режиме эмуляции проекта на персональном компьютере укажите адрес 127.0.0.1. Если необходимо просматривать значения узлов проекта, выполняющегося на контроллере или панели оператора, укажите IP-адрес контроллера или панели.
3. Нажать кнопку «Подключение».

Поле подключения к среде исполнения нужно добавить в окно "Наблюдение" узлы, значения которых необходимо наблюдать. Такие узлы должны иметь возможность чтения или записи значений – это регистры Modbus и другие узлы типа "Источник", сигналы.

Для добавления узлов в окно "Наблюдение" необходимо нажать на нужном узле в дереве проекта правой кнопкой мыши и выбрать команду «Добавить в окно

наблюдения». Выбранные узлы будут отображаться в окне «Наблюдение» вместе со считанными значениями.

### 3.3 Мониторинг событий

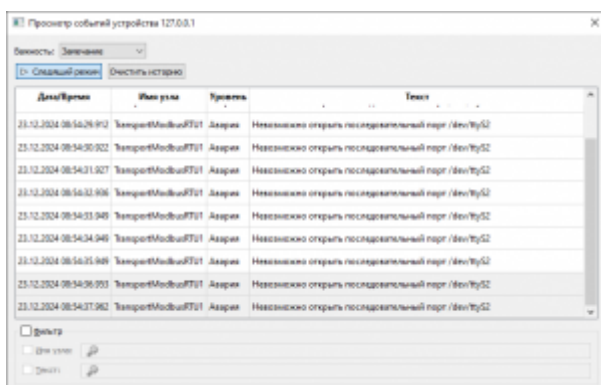


Рисунок 15 – Окно просмотра событий

Мониторинг событий позволяет наблюдать за исполнением проекта, работающего в устройстве. При включении мониторинга пользователь получает информацию о возникающих нефатальных ошибках и предупреждениях, возникающих в ходе исполнения проекта со следующими признаками:

- метка времени, соответствующая времени возникновения события;
- степень важности события;
- тип события - информационное или события активирования, деактивирования и квитирования тревог;
- имя узла, сформировавшего событие.
- текст события.

#### 3.3.1 Использование мониторинга

Для старта мониторинга необходимо подключиться к нужному устройству способом, рассмотренным в разделе [#Мониторинг сигналов](#), а затем вызвать команду «Устройство → Просмотр событий». В открывшемся окне будут выводиться все события, регистрируемые устройством.

С помощью выпадающего списка в левом верхнем углу окна можно выбрать минимальный уровень выводимых событий.

#### 3.3.2 Фильтрация событий

Функция доступна начиная с версии *AgavaSCADA/AgavaPLC 1.2.56+, 1.5.25+*.

В нижнем левом углу окна расположена область настройки фильтра событий, позволяющая фильтровать выводимые события по имени узла и тексту событий. При использовании фильтра возможно задание фильтра в виде:

- простой строки, при этом в список попадают события, содержащие полное совпадение заданной строки; при задании строки возможно в(ы)ключение признака

учета регистра символов в строке;

- регулярного выражения, при этом в список попадают события, удовлетворяющие введенное выражение. Правила составления регулярных выражений описаны на странице [Использование регулярных выражений](#).

### 3.4 Отладка в задаче ПЛК

*Данный раздел актуален для версии AS 1.2.61 и старше. В версиях AS 1.6+ используется современный пошаговый отладчик.*

Задача ПЛК (см. раздел [#Задача ПЛК](#)) позволяет проводить тестирование проектов с помощью режима симулятора (на компьютере, без использования контроллера) или онлайн отладки (на контроллере).

#### 3.4.1 Управление отладкой

Управление режимом симуляции и онлайн отладкой осуществляется с помощью панели инструментов задачи ПЛК (см. рисунок 16).

Панель инструментов содержит следующие элементы:

- Кнопка «Запустить симуляцию».
- Кнопка «Сделать шаг на один цикл».
- Кнопка «Приостановить симуляцию» - ставит выполнение задачи на паузу.
- Кнопка «Остановить симуляцию» - сбрасывает состояние всех блоков к их состоянию по умолчанию.
- Список выбора периода выполнения задачи.
- Кнопка «Показать окно «Наблюдение»».

#### 3.4.2 Режим симулятора

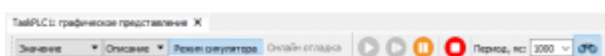


Рисунок 16 - Панель инструментов задачи ПЛК

Режим симулятора позволяет провести тестирование работы алгоритмов проекта на компьютере. С помощью симуляции можно выявить ошибки в работе алгоритмов проекта до его загрузки в контроллер.

Управление симуляцией осуществляется через команды на панели инструментов задачи ПЛК.

Для перехода в режим симуляции нажмите кнопку «Режим симулятора» на панели управления задачи ПЛК.

После активации режима симуляции, для изменения значения входа дважды щелкните на нем левой кнопкой мыши и выберите нужный тип и значение. Текущие значения всех блоков, в соответствии с работой программы, будут отображаться в реальном времени.

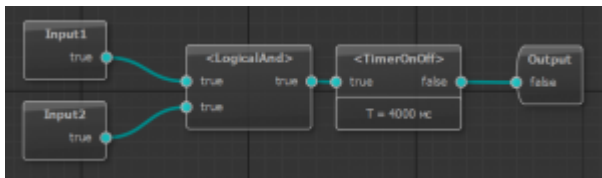


Рисунок 17 – Вид блоков задачи в режиме симулятора

Для выхода из режима симуляции повторно нажмите кнопку «Режим симуляции» на панели управления задачи ПЛК.

### 3.4.3 Онлайн отладка

Режим онлайн отладки позволяет провести тестирование проекта, загруженного и исполняемого в контроллере. Для перехода в режим онлайн отладки загрузите конфигурацию в контроллер, подключитесь к нему, после чего нажмите кнопку «Онлайн отладка» на панели инструментов задачи ПЛК.



Рисунок 18 – Вид блоков задачи в режиме онлайн отладки

На рисунке 18 изображен вид задачи ПЛК в режиме онлайн отладки проекта, запущенного на МПР-60 с установленными submodule DI (входы) и R (выходы).

Для онлайн отладки на панели инструментов недоступны кнопка «Приостановить симуляцию» и список выбора периода.

## 4 Компиляция проекта

Перед загрузкой проекта в контроллер или запуском эмуляции проекта необходимо откомпилировать проект и получить файл конфигурации, предназначенный для исполнения средой исполнения, установленной в контроллере или на персональном компьютере (в случае разработки системы диспетчеризации). Для этого необходимо вызвать команду меню "Проект->Сохранить конфигурацию", выбрать папку проекта и нажать кнопку "Сохранить".

### 4.1 Проверка проекта



Рисунок 19 – Окно «Ошибки»

Перед сохранением проекта желательно выполнить его проверку на наличие ошибок с помощью команды «Проект->Проверить проект». После проверки обнаруженные ошибки будут отображены в окне «Ошибки».

Если отображение окна «Ошибки» выключено, его можно включить командой «Вид->Показать окно ошибок».

Для облегчения исправления обнаруженных ошибок окно поддерживает выполнение некоторых команд над обнаруженными ошибками, вызывать которые можно из контекстного меню, отображаемого щелчком правой кнопки мыши на строке с ошибкой в окне. Контекстное меню позволяет вызывать команду «Показать узел в дереве конфигурации», выполняющую соответствующее действие.

Также щелчок левой кнопкой мыши на строке с ошибкой открывает окно «Свойства» и дает возможность изменить в нем все свойства узла, содержащего ошибку.

#### 4.1.1 Установка пароля на файл конфигурации

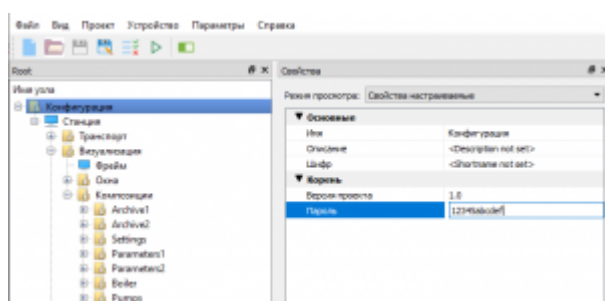


Рисунок 13 - Установка пароля

Файл конфигурации, загружаемый в контроллер или панель оператора, можно защитить паролем, чтобы предотвратить его несанкционированное изменение.

Для установки пароля на файл конфигурации необходимо выбрать в дереве проекта узел «Конфигурация» и отредактировать его свойство «Пароль». Допускается использование любых цифр, букв и спецсимволов.

В дальнейшем при открытии файла конфигурации в среде разработки будет появляться окно с запросом пароля.

## 5 Загрузка проекта в устройство

Загрузка проекта в целевое устройство может быть осуществлена несколькими способами:

- загрузка из среды разработки;
- загрузка с помощью USB-Flash накопителя (для панелей оператора);
- загрузка через браузер (для МПР-60).

### 5.1 Загрузка из среды разработки

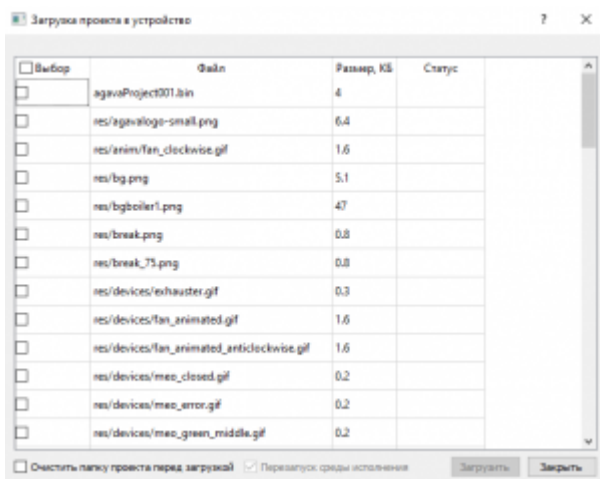


Рисунок 20 – Окно «Загрузка проекта в устройство» после подключения к контроллеру

Для загрузки проекта из среды разработки необходимо произвести следующие действия в указанном порядке:

1. Создайте файл конфигурации.
2. Выполните подключение целевого устройства к ПК по сети Ethernet.
3. После подключения вызовите в среде разработки команду «Проект->Загрузка проекта в устройство».
4. В открывшемся окне будет отображен список файлов проекта, доступных для загрузки в целевое устройство.
5. Выберите необходимые файлы и нажмите кнопку «Загрузить». После нажатия кнопки начнется процесс загрузки файлов, при этом в нижней части окна будет меняться состояние полосы прогресса, а также для каждого файла в таблице будет отображен результат загрузки.
6. После окончания загрузки среда исполнения в целевом устройстве будет перезапущена и загруженный проект будет запущен.

## 5.2 Загрузка через USB-Flash (для ПО-40 и ПО-50)

Для загрузки проекта в устройство с помощью USB-Flash накопителя необходимо произвести следующие действия в указанном порядке:

1. Подготовьте USB-Flash накопитель, предварительно отформатировав его в файловую систему FAT32.
2. Вызовите в среде разработки команду «Подготовка архива для обновления» в меню «Проект».
3. Сохраните архив в корень USB-Flash накопителя.
4. Перейдите в системную утилиту, перезагрузив панель оператора, при загрузке во время однократного звукового сигнала нажмите на экран и удерживайте до повторного звукового сигнала. Введите пароль: 111111.
5. Перейдите на вкладку «Обновление», затем подключите USB-Flash накопитель

через переходник USB-OTG, выберите из списка загрузочный архив, нажать кнопку «Обновить». Дождитесь сообщения «Обновление успешно» и нажмите кнопку «ОК».

6. Перейдите на вкладку «Перезапуск» и нажмите кнопку «Перезапуск».

После перезагрузки панели загруженный проект будет запущен.

### **5.3 Загрузка через браузер (для МПР-60)**

Для загрузки проекта через браузер необходимо произвести следующие действия в указанном порядке.

1. Вызовите в среде разработки команду «Подготовка архива для обновления» в меню «Проект».
2. Сохраните архив.
3. Откройте системную утилиту, введя IP-адрес контроллера в адресную строку браузера.
4. Перейдите на вкладку «Загрузка ПО». Рисунок 71 - Вкладка системной утилиты «Загрузка ПО»
5. Выберите на компьютере архив с проектом, который необходимо загрузить в контроллер.
6. Нажмите кнопку «Загрузить».
7. Дождитесь окончания загрузки файла и обновления проекта.

## **6 Отладка проекта**

---

*Данный функционал доступен в версиях AS 1.6+. Для версий AS 1.2 используйте мониторинг, отладку в задачах ПЛК.*

### **6.1 Локальная отладка**

### **6.2 Удаленная отладка**

---

Источник — [https://docs.kb-agava.ru/index.php?title=Универсальная\\_среда\\_разработки\\_Agava&oldid=3471](https://docs.kb-agava.ru/index.php?title=Универсальная_среда_разработки_Agava&oldid=3471)

---

Эта страница в последний раз была отредактирована 27 марта 2026 в 14:49.