

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
"ТСА-Сервис"



ОКПД 2 26.51.70.190



УТВЕРЖДАЮ
Генеральный директор
ООО «ТСА-Сервис»
_____ Петров С.В.
«01» июня 2021 г.

Комплекс программно-технический Квинт-6

SCADA-система «Квинтегратор»

Графический редактор Графит
Руководство пользователя
ПФДИ.421457.009 И3.5

Москва
2021

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Содержание

1	Введение	7
1.1	Назначение и структура документа.....	7
1.2	Необходимое программное обеспечение.....	7
1.2.1	Квинт.....	7
1.2.2	Borland Database Engine.....	7
1.2.3	Редактор растровых изображений.....	7
1.2.4	Редактор звуковых файлов.....	7
1.3	Запуск Графита.....	7
1.3.1	Рекомендации.....	8
1.4	Краткие сведения о структуре проекта Квинта.....	8
2	Рисование	9
2.1	Изображения.....	9
2.2	Открытие изображений.....	10
2.3	Создание нового изображения.....	10
2.4	Удаление изображения.....	10
2.5	Задание свойств изображения.....	10
2.6	Редактирование векторных изображений.....	10
2.6.1	Свойства векторных изображений.....	10
2.6.2	Структура изображения.....	11
2.6.3	Добавление элементов.....	11
2.6.4	Выделение элементов.....	11
2.6.5	Геометрические преобразования элементов.....	12
2.6.6	Изменение формы элементов.....	13
2.6.7	Координаты.....	13
2.6.8	Атрибуты элементов.....	13
2.6.9	Атрибуты общего назначения.....	13
2.6.10	Атрибуты раскраски.....	14
2.6.11	Атрибуты текста.....	14
2.6.12	Специальные атрибуты.....	14
2.6.13	Точки соединения.....	15
2.6.14	Контекстное меню элемента.....	15
2.6.15	Перестановка элементов.....	16
2.6.16	Группировка элементов.....	16

Подп. и дата	Инв. №	Взаим. инв. №	Подп. и дата	Инв. № подл.

Изм	Лист	№ докум	Подп.	Дата
Разраб.		Туркин		
Пров.		Зарипов		
Н.контр		Бочаров		
Утверд.		Петров		

ПФДИ.421457.009 И3.5								
Комплекс программно-технический Квинт-6. SCADA-система «Квинтегратор» Графический редактор Графит. Руководство пользователя.						Лит	Лист	Листов
							2	126
						ООО «ТСА-Сервис»		

2.6.17	Выравнивание элементов.....	17
2.6.18	Регулирование положения.....	17
2.6.19	Регулирование размера.....	17
2.6.20	Использование автоматического растягивания.....	17
2.6.21	Пример создания окна.....	18
2.6.22	Использование клавиатуры для перемещения курсора.....	19
2.7	Редактирование картинок.....	19
2.7.1	Краткое описание основных действий.....	19
2.7.2	Преобразования.....	19
2.7.3	Прозрачность.....	20
2.7.4	Кадры.....	20
2.7.5	Свойства картинок.....	20
3	Типы элементов.....	21
3.1	Основные.....	22
3.1.1	Отрезок.....	22
3.1.2	Прямоугольник.....	22
3.1.3	Эллипс.....	23
3.1.4	Сектор (сегмент, дуга).....	23
3.1.5	Ломаная.....	23
3.1.6	Кривая.....	25
3.1.7	Текст.....	26
3.1.8	Шкала.....	27
3.1.9	Круговая шкала.....	28
3.1.10	Стрелка.....	29
3.1.11	Реквизит.....	29
3.1.12	Дата-время.....	30
3.1.13	Строка состояния.....	30
3.1.14	Отросток.....	31
3.1.15	Список состояний.....	31
3.1.16	Список изображений.....	32
3.1.17	Вставка.....	33
3.2	Групповые.....	35
3.2.1	Общие свойства групповых элементов.....	35
3.2.2	Объединение.....	35
3.2.3	Соединение.....	36
3.2.4	Серия.....	36
3.2.5	Кнопка.....	37
3.2.6	Уровень.....	37
3.2.7	Движок.....	38
3.2.8	Ползунок.....	39

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата

3.3	Дополнительные	40
3.3.1	Общие свойства	40
3.3.2	График	40
3.3.2.1	Назначение	40
3.3.3	Таблица данных из архива	41
3.3.4	Окно ошибок	42
3.3.5	Полис	44
3.3.6	Список параметров	45
3.3.7	Состояния Архивных станций	46
3.3.8	Хроника дискретных событий	47
3.3.9	ActiveX	48
4	Компоновка экрана	49
4.1	Рабочие столы	49
4.1.1	Понятие рабочего стола	49
4.1.2	Редактор рабочих столов	49
4.1.3	Использование рабочих столов	51
4.1.4	Подключение нескольких мониторов	51
4.1.5	Автоматическое масштабирование рабочих столов	51
4.1.6	Переключение рабочих столов оператором	52
4.2	Мнемокадры	52
4.2.1	Основные мнемокадры	52
4.2.2	Плавающие мнемокадры	53
4.2.3	Закрепленные мнемокадры	53
4.3	Прочие окна	54
4.3.1	Окна сигнализации	54
4.3.2	Диалоговые окна	54
4.4	Сохранение состояния рабочего стола	54
4.5	Реализация функций событийной и приборной станций	55
4.5.1	Постановка задачи	55
4.5.2	Подход к решению	55
4.5.3	Создание мнемосхемы и рабочего стола	55
4.5.4	Настройка списков	56
5	Задание динамики	57
5.1	Объекты и их параметры	57
5.1.1	Типы объектов	57
5.1.2	Типы параметров	57
5.1.3	Свойства параметров	57
5.1.4	Нормативные параметры	59

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

5.2	Анимация	59
5.2.1	Понятие аниматора	59
5.2.2	Редактирование аниматоров	60
5.2.3	Аниматоры состояний и наборов бит	61
5.2.4	Аниматоры числовых параметров	62
5.2.5	Аниматоры параметров времени	63
5.3	Управление	63
5.3.1	Понятие рецептора	63
5.3.2	Редактирование рецепторов	64
5.3.3	Рецепторы категории Параметр	66
5.3.4	Рецепторы категории Открыть	70
5.3.5	Рецепторы категории Активное окно	71
5.3.6	Рецепторы "Управление окнами"	72
5.3.7	Рецепторы категории Звук	73
5.3.8	Рецепторы категории Прочие	75
5.4	Встроенная программа	78
5.4.1	Для чего нужна программа	78
5.4.2	Синтаксис	79
5.4.3	Как устроена программа	79
5.4.4	Что может делать программа	80
5.4.5	Как создать программу	81
5.4.6	Подключение динамических библиотек	82
5.4.7	Отладка программ	83
5.5	Отладка динамики	83
5.5.1	Режим обзора	83
5.5.2	Эмулятор	83
6	Дополнительные возможности	85
6.1	Настройки	85
6.1.1	Настройка палитры	85
6.1.2	Настройка стилей текста	86
6.1.3	Копирование настроек	87
6.2	Печать изображений	87
6.2.1	Печать	87
6.2.2	Настройка параметров страницы	87
6.2.3	Печать фона	87
6.3	Импорт и экспорт изображений	87
6.3.1	Копирование изображения из одного проекта в другой	87
6.3.2	Замена содержимого изображения	88
6.3.3	Сохранение фрагмента изображения в виде файла	89
6.3.4	Копирование рисунков из других программ	89
6.3.5	Экспорт рисунков в другие программы	89

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата

6.4	Проверка изображений.....	90
6.4.1	Для чего нужна проверка	90
6.4.2	Примеры ошибок.....	90
6.4.3	Проверка одного изображения	90
6.4.4	Проверка всех изображений	91
7	Справочник.....	92
7.1	Описание пользовательского интерфейса	92
7.1.1	Главное меню.....	92
7.1.2	Панель инструментов для редактирования изображения.	95
7.1.3	Панель инструментов для редактирования элементов изображения	96
7.2	Справочник по встроенной программе	98
7.2.1	Константы	98
7.2.2	Функции.....	98
7.2.3	Доступ к элементу изображения	111
7.2.4	Доступ к параметрам объектов	115
7.2.5	Доступные поля базы данных (БД) Квинта.....	117
7.2.6	Синтаксические диаграммы.....	121
	Лист регистрации изменений.....	126

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

меняться оператором. Если размер окна меньше размера изображения, то в окне появляются полосы прокрутки.

Когда изображение открыто в Графите, его размер отображается с помощью **маркера размера** черного цвета (рисунок 1). Перемещая маркер, можно менять размер изображения.

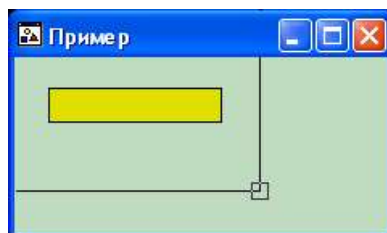




Рисунок 1 - Задание размера изображения


При открытии изображения в Графите размер окна устанавливается равным размеру изображения. В дальнейшем можно подогнать размер окна под размер изображения с помощью пункта **Подогнать** системного меню (вызывается по кнопке  слева от слова **Пример**, рисунок 1).

2.6.2 Структура изображения

Каждое изображение состоит из **элементов**. (Для элемента изображения иногда еще используется термин **примитив**). В процессе редактирования изображения можно добавлять или удалять его элементы, изменять свойства отдельных элементов и менять их порядок. Порядок элементов определяет последовательность их вывода на экран и является существенным, если элементы перекрываются.

Состав изображения иногда удобно наблюдать в специальном **окне списка элементов**, которое вызывается нажатием кнопки . Элементы, расположенные в списке выше, рисуются на экране первыми.

Все элементы делятся на **одиночные** и **групповые**. Одиночный элемент изображения – это такой элемент, внутреннее содержимое которого не может быть изменено в процессе редактирования данного изображения. Групповой элемент всегда состоит из других элементов, и его содержимое доступно для редактирования.

Чтобы получить доступ к содержимому группового элемента, нужно выделить этот элемент и нажать кнопку  или произвести двойной щелчок по соответствующей строчке в окне со списком элементов.

Групповой элемент может содержать как одиночные, так и групповые элементы. Таким образом, структура элементов в изображении может быть иерархической.


Каждый элемент принадлежит к определенному типу. **Тип элемента** определяет его способ рисования, набор свойств и многие другие особенности. О специфических особенностях каждого типа подробно рассказано в разделе «Типы элементов».

Каждому элементу можно назначить **имя**. Задание имени не является обязательным; имя представляет собой дополнительное описание элемента. Имена элементов отображаются в окне со списком элементов. В настоящее время имена используются только для различения элементов в списке. Для задания имени выберите первый пункт контекстного меню элемента.

2.6.3 Добавление элементов

Чтобы добавить к изображению элемент, нажмите кнопку с символом нужного элемента в панели инструментов. Затем нарисуйте элемент в окне изображения, следуя инструкциям рисования для данного типа элементов. Эти инструкции приведены в разделе 3. Пока рисование элемента не закончено, кнопка в панели инструментов остается нажатой. После окончания рисования кнопка автоматически отжимается, и Графит переходит в режим выделения элементов.

2.6.4 Выделение элементов

Режим выделения элементов устанавливается сразу при открытии окна. В него всегда можно перейти, нажав кнопку  на панели инструментов.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Чтобы произвести какую-нибудь операцию с элементом (скажем, удалить), нужно его вначале выделить. Одновременно может быть выделено несколько элементов изображения, это удобно для проведения групповых операций.

Выделенный элемент отмечается 8-ю квадратными **маркерами**, расположенными по краям элемента: 4 **угловых** маркера служат для изменения размера элемента с сохранением его пропорций, 4 **боковых** маркера – для изменения размера без сохранения пропорций.

Примечание - Угловые маркеры могут не отображаться, если ширина или высота элемента малы. Если выделено несколько элементов, маркеры отмечают описанный около них прямоугольник.


Чтобы **выделить элемент**, щелкните левой кнопкой мыши в любой точке, принадлежащей элементу. Все прочие элементы при этом станут невыделенными. (Если элемент уже был в числе выделенных, выделение прочих элементов сохранится). Если элемент не имеет заполнения, нужно щелкнуть вблизи его контура. Поскольку попасть мышью точно в линию контура довольно сложно, вводится допуск, величину которого можно задать в настройках. Если курсор мыши попадает в элемент, он изменяет свой вид.

Для **выделения нескольких элементов** щелкните по ним последовательно, удерживая нажатой клавишу **Shift**. При нажатой клавише **Shift** щелчок мыши по элементу инвертирует его выделение, а на выделение других элементов не влияет.

Если требуется **выделить элементы, находящиеся в одной области изображения**, обведите их прямоугольной рамкой. Для этого нажмите левую кнопку мыши в точке, не содержащей элементов, и переместите мышь в противоположный угол рамки, удерживая кнопку. После отпущения кнопки мыши будут выделены все элементы, целиком лежащие внутри данной рамки.

Для **выделения всех элементов** изображения удобно использовать команду меню **Правка/Выделить все**.

Чтобы **снять выделение** со всех элементов достаточно щелкнуть мышью в точке, не содержащей элементов.

В некоторых случаях выделить элемент мышью практически невозможно (например, если элемент целиком скрыт под другим элементом, визуально сливается с ним или находится вне видимой части изображения). Тогда для выделения можно использовать окно со списком элементов, вызываемое кнопкой . Выделение элементов в этом списке и в изображении синхронизировано.

Иногда требуется выделить все элементы, обладающие определенными свойствами (например, все красные прямоугольники). Для этого можно использовать диалоговое окно, вызываемое из меню **Правка/Выделить по атрибуту**.

2.6.5 Геометрические преобразования элементов

В таблице 3 видно, как Графит позволяет выполнять всевозможные геометрические (аффинные) преобразования элементов.

Таблица 3 – Геометрические преобразования элементов

№	Преобразование	Режим	Действие
1	Перемещение		Перемещать за любую точку элемента
2	Масштабирование (с сохранением пропорций)		Перемещать угловой маркер
3	Сжатие и растяжение по вертикали или по горизонтали		Перемещать боковой маркер
4	Отражения относительно вертикальной или горизонтальной оси	Любой	Кнопки  и 
5	Повороты на 90° и 180°.	Любой	Кнопка 
6	Поворот на произвольный угол		Перемещать угловой маркер. Центр вращения можно задать с помощью центрального маркера.
7	Скашивание		Перемещать боковой маркер


Важно понимать, что не каждый элемент допускает любое преобразование. Так, для прямоугольника можно выполнить операции 1 – 5, для текста 1, 2, 3 и 5, для картинки – 1, 4 и 5.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Если хотя бы для одного их выбранных элементов преобразование недоступно, соответствующий маркер рисуется незаполненным.

2.6.6 Изменение формы элементов

Геометрические преобразования воздействуют на все элементы одинаково, изменяя их положение и размер. Изменение формы является специфическим для каждого типа элементов, позволяя, например, изменить угол раствора сектора.

Инструмент изменения формы  позволяет перемещать контрольные точки элемента с помощью мыши. Количество и расположение контрольных точек зависит от типа элемента, и описано в разделе 3. Контрольные точки отображаются только для одного выделенного элемента. Щелкнув по другому элементу, можно сделать выделенным его.

2.6.7 Координаты

Этот раздел носит более специальный характер, и может быть пропущен при первом чтении.

Каждый элемент изображения «помнит» свой размер, позицию и, если нужно, другие геометрические координаты. Это – так называемые **мировые** координаты. Они не изменяются, когда пользователь прокручивает изображение или изменяет его масштаб. В этом их отличие от **оконных** координат, которые выражают расстояние от края окна в пикселях. Оконные координаты отсчитываются от верхнего левого угла окна. Пересчет мировых координат элемента в оконные производится при выводе изображения на экран.

Мировые координаты в Графите выражаются, как и оконные, целыми числами. В исходном состоянии окна мнемосхемы значения мировых координат ровно в 8 раз больше оконных. Если три раза выполнить команду **Вид / Увеличить**, то расстояния в оконных и мировых координатах будут совпадать.

Операторская станция может производить автоматическое масштабирование рабочих столов. При этом возникают дополнительные множители для пересчета из мировых координат в экранные.

Мировые координаты используются во встроенной программе и в некоторых диалоговых окнах настройки.

2.6.8 Атрибуты элементов

Атрибут – это такое свойство элемента, которое может изменяться, как в процессе редактирования, так и в оперативном режиме.

Набор атрибутов элемента зависит от типа элемента. Например, прямоугольник имеет атрибут «цвет заполнения», а отрезок – нет.

Один и тот же атрибут может иметь смысл для элементов разного типа, например, «толщина контура» есть и у прямоугольника, и у отрезка.

Атрибуты используются:

- при настройке внешнего вида элемента в редакторе;
- для изменения вида элементов в оперативном режиме с помощью аниматоров;
- для изменения вида элементов в оперативном режиме с помощью встроенной программы;
- для поиска элементов (элемент меню **Правка / Выделить по атрибуту**).

Чтобы изменить атрибут элемента, следует вызвать **диалоговое окно задания атрибутов**. Это можно сделать 2-мя способами:

- 1 Щелкнув по элементу правой кнопкой мыши, выбрать в выпавшем меню пункт **Атрибуты**.
- 2 Выделить один или несколько элементов и из главного меню выбрать пункт **Преобразования / Атрибуты**.

Наиболее часто используемые атрибуты - «**Цвет заполнения**» и «**Цвет контура**» - можно изменять, не вызывая диалоговое окно. Щелчком левой кнопки мыши по палитре в главном окне можно изменить цвет заполнения выделенных элементов, щелчком правой кнопки – цвет контура.

2.6.9 Атрибуты общего назначения

Атрибуты общего назначения (таблица 4) применимы к элементам к элементам любого типа. Они задаются в закладке **Общие** диалога **Задание атрибутов**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						13

Таблица 4 – Атрибуты общего назначения

Атрибут	Назначение	Значение по умолчанию
Видимость	Определяет видимость элемента в оперативном режиме. (В режиме редактирования видимы все элементы).	Да
Запрет управления	Если установлен запрет управления, то в оперативном режиме элемент не реагирует на события мыши и клавиатуры.	Нет
Растягивание	Определяет, как будет изменяться размер элемента при изменении размера окна, содержащего этот элемент. См. раздел «Использование автоматического растягивания»	Нет
Вызов меню рецепторов по левой кнопке	Если в оперативном режиме щелкнуть по элементу правой кнопкой мыши, то будет выведено меню, содержащее названия рецепторов этого элемента. Если данный атрибут имеет значение «Да», то при щелчке левой кнопкой также будет выведено это меню.	Нет

2.6.10 Атрибуты раскраски

Атрибуты раскраски (таблица 5) определяют вид контура и заполнения элементов. Они задаются в закладках **Заполнение**, **Контур** и **Фон** диалога **Задание атрибутов**. Значения по умолчанию для этих атрибутов можно настроить с помощью меню **Вид / Цвет по умолчанию**.

Таблица 5 - Атрибуты раскраски

Атрибут	Назначение	Значение по умолчанию
Заполнение	Цвет	Цвет внутренней части элемента. При использовании штрихованного заполнения означает цвет штриховых линий.
	Частота мигания	Частота мигания внутренней части элемента.
	Шаблон	Вид штриховки заполнения
Контур	Цвет	Цвет контура элемента. При использовании пунктирного контура означает цвет пунктира.
	Частота мигания	Частота мигания контура
	Шаблон	Толщина линии и вид пунктира. Шаблон «Тонкий» означает, что контур будет иметь толщину в 1 пиксель при любом увеличении окна.
Фон	Цвет	Цвет промежутков между линиями штриховки и пунктиром контура. Фон не виден при сплошном заполнении и сплошном контуре.
	Частота мигания	Частота мигания фона

2.6.11 Атрибуты текста

Атрибуты текста (таблица 6) применимы к элементам, содержащим текст. Они задаются в закладках **Текст**, и **Формат** диалога **Задание атрибутов**. Значения по умолчанию для атрибутов из закладки **Текст** можно настроить с помощью меню **Вид / Цвет по умолчанию**.

Таблица 6 – Назначение атрибутов текста

Атрибут	Назначение	Значение по умолчанию
Цвет текста	Цвет символов текста. Фон, на котором рисуются символы, определяется атрибутами фона.	0-й элемент палитры
Частота мигания текста	Частота мигания символов текста	0 Hz (не мигает)
Стиль текста	См. 6.1.2.	
Выравнивание текста	Определяет расположение строк относительно прямоугольной области элемента. Выравнивание не имеет смысла, если для текста установлен флажок Автоматический размер . Для многострочного текста работает только выравнивание по горизонтали.	Влево вверх
Формат	См. 3.1.7.	

2.6.12 Специальные атрибуты

Специальные атрибуты (таблица 7) имеют смысл только для отдельных типов элементов. Они задаются в специальных диалогах настройки для этих элементов или во встроенной программе.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Таблица 7 – Назначение специальных атрибутов

Атрибут	Назначение	Значение по умолчанию
Номер кадра	Эти атрибуты определены для вставленных в изображение картинок.	0
Период смены кадра	См. 2.7.	0 сек
Уровень шкалы	Этот атрибут определен для элементов типа «Уровень», «Движок» и «Стрелка». Эти элементы в разной форме отображают значение некоторой аналоговой величины. Эта величина может быть задана как параметр объекта, либо непосредственно указана во встроенной программе. Значение должно меняться в пределах [0, 1].	

2.6.13 Точки соединения

При рисовании технологических схем часто приходится изображать элементы, соединенные линиями. Примерами могут служить два агрегата, соединенных трубопроводом, или схематичная связь между двумя объектами диаграммы (рисунок 2). Для экономии времени в ходе дальнейшего редактирования в Графите предусмотрена следующая возможность: при изменении положения связанного элемента все связи отслеживают это изменение:

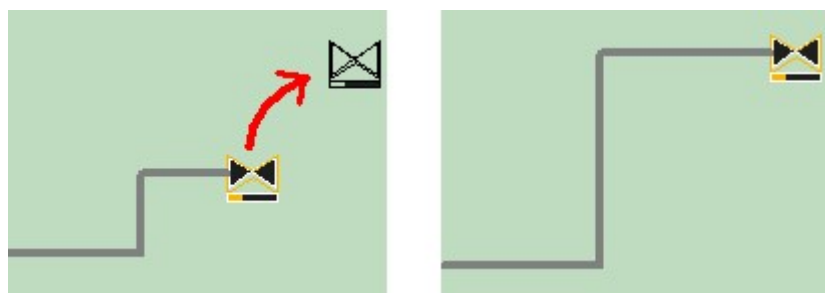



Рисунок 2 – Пример рисования технологических схем

Эта возможность реализована с помощью следующих понятий:

- **соединяющий элемент** – элемент, который может присоединяться к другим элементам;
- **точка соединения** – точка элемента, в которой к нему может присоединяться соединяющий элемент.

Соединяющими элементами в Графите могут быть ломаные и кривые линии.

Многие типы элементов имеют набор характерных точек соединения. Например, у прямоугольника точками соединения являются середины его сторон. Характерные точки соединения для каждого типа описаны в разделе 3. Кроме того, для групповых элементов и вставок можно задавать дополнительные точки соединения с помощью элемента «отросток».

Точки соединения отображаются при рисовании соединения и при изменении формы соединения инструментом . Если подвести курсор к элементу, имеющему точки соединения, то все точки соединения этого элемента будут помечены крестиками, а та точка, возле которой находится курсор, жирным квадратиком. При отпускании кнопки мыши соединяющий элемент пристыковывается к точке, отмеченной квадратиком.

Соединяющие элементы отслеживают все изменения положения и размера тех элементов, к которым они присоединены. Если выделить и скопировать несколько элементов вместе со связями, то копия сохранит информацию о своих внутренних связях. Сказанное распространяется только на режим редактирования; в оперативном режиме связи не отслеживаются.

2.6.14 Контекстное меню элемента

Контекстное меню элемента вызывается щелчком правой кнопкой мыши по любой точке этого элемента. Если элемент не был выделен, то он станет выделенным, а все остальные элементы – невыделенными. Если этот элемент был в числе выделенных, то выделение не изменится.

Контекстное меню обычно состоит из двух частей: общей (сверху) и специальной (снизу). Пункты меню из общей части всегда одинаковы, в то время как количество и назначение пунктов из специальной части зависят от типа элемента изображения. Пункты специальной части меню описаны для каждого типа элемента в разделе 3. Пункты общей части перечислены в таблице 8.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	


Таблица 8 – Назначение контекстного меню элемента

<Обозначение элемента>	Вызывает диалог задания имени элемента. Обозначение элемента включает в себя имя и общую характеристику элемента.
Атрибуты	Вызывает окно задания атрибутов элемента.
Выравнивание	Вызывает окно задания выравнивания.
Программа	Вызывает окно редактирования встроенной программы .
Вырезать	Копирует выбранные элементы в буфер обмена и удаляет их с изображения
Копировать	Копирует выбранные элементы в буфер обмена
Удалить	Удаляет выбранные элементы с изображения
Удвоить	Добавляет к изображению копию выделенных элементов. Копия смещается относительно оригинала на величину, задаваемую в настройке Графит/Смещение при копировании

2.6.15 Перестановка элементов

Элементы изображения образуют упорядоченный список. При добавлении нового элемента он помещается в конец этого списка.

Порядок элементов определяет последовательность их вывода на экран. Каждый следующий элемент рисуется поверх уже нарисованных элементов. Если 2 элемента расположены в разных частях изображения, то их взаимный порядок несущественен. Если же области элементов пересекаются, то элемент, идущий позже, перекроет (частично или полностью) элемент, идущий раньше.

Порядок элементов наглядно виден в окне «список элементов», вызываемом кнопкой . Элементы в конце списка будут перекрывать на изображении элементы из начала списка.

Перестановкой элементов можно добиться, чтобы они следовали в нужном порядке. В таблице 9 перечислены реализованные в Графите операции перестановки.

Таблица 9 – Операции перестановки


Операция	Описание	Способ вызова	
		Панель инструментов	Окно со списком элементов
Вперед	Выделенные элементы становятся последними в списке (верхними на изображении)		
Назад	Выделенные элементы становятся первыми в списке (нижними на изображении)		
Вперед на один	Номер каждого выделенного элемента увеличивается на 1		
Назад на один	Номер каждого выделенного элемента уменьшается на 1		


2.6.16 Группировка элементов

Графит позволяет сгруппировать вместе несколько элементов изображения. После группировки они представляются в списке как один элемент, называемый **объединением**.

Группировка обычно применяется для следующих целей:

- 1 Фиксация взаимного расположения элементов – все операции с объединением (выделение, преобразования и др.), будут применяться сразу ко всем входящим в него элементам.
- 2 Организация логической структуры изображения – группировка делает структуру изображения иерархической.
- 3 Задание особенностей прорисовки объединения (3.2.2)
- 4 Установка аниматоров и рецепторов для объединения.

Чтобы сгруппировать несколько элементов, выделите их (как описано в 2.6.4) и нажмите кнопку  (или вызовите пункт меню **Преобразования / Объединить**). Полученный элемент «Объединение» займет в списке элементов позицию, соответствовавшую первому объединяемому элементу.

Чтобы получить доступ к содержимому объединения, нужно его выделить и нажать кнопку  или произвести двойной щелчок по соответствующей строчке в окне со списком элементов. Элементы

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

разного рода списки; обычно они имеют собственные линейки прокрутки. Для таких элементов рекомендуется задавать автоматическое растягивание.

Автоматическое растягивание – это изменение (по определенным законам) размера или положения элемента при изменении размера окна.

Растягивание элемента настраивается в закладке **Общие** диалога настройки атрибутов элемента и имеет несколько вариантов (таблица 12).

Таблица 12 – Варианты настроек для растягивания элементов

Нет	Элемент не меняет размер и положение при изменении размеров окна. Это – значение по умолчанию.
Вверх	Элемент растягивается по всей ширине окна, примыкая к его верхней границе. Высота элемента остается неизменной.
Вниз	Элемент растягивается по всей ширине окна, примыкая к его нижней границе. Высота элемента остается неизменной.
Влево	Элемент растягивается по всей высоте окна, примыкая к его левой границе. Ширина элемента остается неизменной.
Вправо	Элемент растягивается по всей высоте окна, примыкая к его правой границе. Ширина элемента остается неизменной.
Во все стороны	Элемент растягивается по всей области, не занятой другими элементами, для которых установлено автоматическое растягивание.

Если размер окна меньше размера изображения, то элементы растягиваются по размеру изображения. Это позволяет ограничить минимальный размер растягиваемых элементов.

Изображение может содержать несколько элементов с автоматическим растягиванием (рисунок 3). В этом случае каждый такой элемент, начиная с первого, будет растягиваться по описанному закону, оставляя для следующих за ним элементов незанятую им площадь.



Рисунок 3 – Пример автоматического растягивания элементов

Если элемент, для которого задано автоматическое растягивание, входит в состав объединения, то этот элемент будет растягиваться не по размеру окна, а по размеру прямоугольника, описанного около объединения.

2.6.21 Пример создания окна

Создадим окно, изображенное на рисунке 4. Окно содержит список ошибок, размер которого изменяется при изменении размера окна, а снизу находится панель фиксированной высоты с кнопкой.

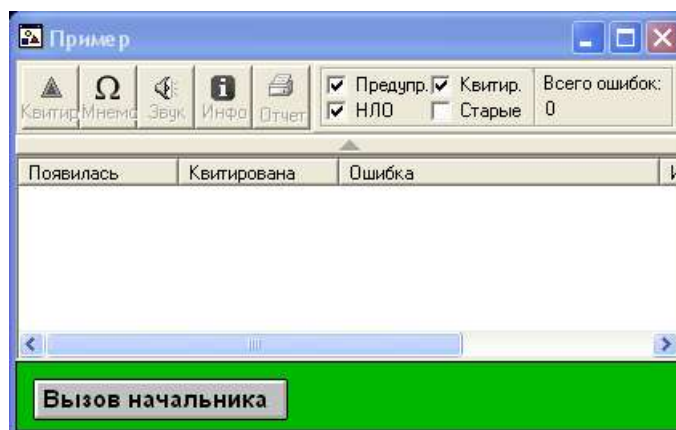


Рисунок 4 – Пример окна для вывода списка ошибок

Инв. № подл.	
Подп. и дата	
Взаим. инв. №	
Инв. №	
Подп. и дата	

Для создания окна произведите последовательно следующие действия:

- 1 Создайте мнемосхему с именем «Пример».
- 2 Добавьте зеленый прямоугольник нужной высоты.
- 3 Добавьте кнопку и сгруппируйте ее с зеленым прямоугольником.
- 4 Для полученного объединения задайте автоматическое растягивание **Вниз**.
- 5 Добавьте элемент **Окно ошибок** (из списка дополнительных элементов).
- 6 Задайте для него автоматическое растягивание **Во все стороны**.
- 7 Изменяя размер окна, определите, при каком минимальном размере список смотрится нормально. Установите соответствующий размер изображения. В данном примере он равен примерно 450x200.

2.6.22 Использование клавиатуры для перемещения курсора

Используя мышь, достаточно трудно перемещать курсор на малые расстояния. Графит позволяет имитировать такие перемещения с помощью клавиатуры. Нажатие клавиш со стрелками перемещает курсор в соответствующем направлении на величину, задаваемую в настройке **Графит/Смещение по стрелкам**. Значение по умолчанию – 1.

2.7 Редактирование картинок

Картинкой в Графите называется растровое изображение. В отличие от векторных, растровые изображения хранят не набор геометрических элементов, а прямоугольную таблицу точек (пикселей).



Для целей статической графики картинки более универсальны – они не ограничены предоставляемым набором элементов, могут содержать любые оттенки и переходы цветов. С другой стороны, возможности задания динамики для картинок очень ограничены. Кроме того, готовое векторное изображение проще редактировать.


Оператору картинки могут быть представлены только в виде встроенных в другие окна элементов. В частности, можно использовать картинку как подложку мнемосхемы.

Графит не располагает инструментарием редактирования растровых изображений. Поэтому для работы может понадобиться редактор растровой графики, работающий с файлами BMP. Можно использовать входящий в состав Microsoft Windows редактор «Paint».

2.7.1 Краткое описание основных действий




Вначале следует нарисовать нужную картинку в растровом редакторе или использовать готовый файл с растровой графикой в формате BMP, JPG, TIF, PNG и т. д.. В редакторе картинку нужно выделить и скопировать ее буфер обмена.

Затем нужно превратить эту картинку в документ Графита. Для этого выберите пункт главного меню **Файл / Открыть / Картинка**. В открывшемся диалоговом окне нажмите кнопку **Новое**, введите произвольное имя картинки и нажмите кнопку **ОК**. В рабочей области Графита откроется пустое окно с введенным именем. Выберите пункт главного меню **Правка / Вставить** или нажмите кнопку . Картинка появится в окне редактирования. Сохраните ее, нажав кнопку .

Теперь эту картинку можно встраивать в мнемосхемы и другие векторные изображения Графита. Откройте мнемосхему, выберите инструмент вставки  и в выпадающем меню укажите тип **Картинка**. В открывшемся диалоговом окне выберите созданную картинку. Установите ее на нужное место мнемосхемы.


Одну и ту же картинку можно встраивать многократно в различные изображения. Изображения сохраняют связь с исходной картинкой; при ее изменении все встроенные копии также изменятся.

2.7.2 Преобразования


Встроенную копию картинки можно поворачивать на 90° и отражать вертикально или горизонтально. Таким образом, для получения изображений стрелок ,  и  достаточно одной исходной картинки.


Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

2.7.3 Прозрачность

Часто бывает необходимо встраивать картинку в изображения с разным цветом фона. Если стрелка  была нарисована на белом фоне, то при встраивании в изображение с цветным

фоном она будет выглядеть неадекватно: 

Чтобы добиться нужного результата, при редактировании картинки в растровом редакторе залейте те области, которые должны быть прозрачными, каким-нибудь неиспользуемым цветом, например, фиолетовым: . (Белый цвет не подходит, так как он используется внутри стрелки). Перенесите картинку в Графит через буфер обмена. Раздвиньте рамку окна редактирования. Картинка в нем

выглядит так: 

«Шашечки» снаружи картинки обозначают, что картинка непрозрачная. Т.е. при встраивании она всегда будет иметь фиолетовую подложку.

Теперь выполните двойной щелчок левой кнопкой мыши по любой фиолетовой точке картинки.

Окно редактирования примет вид: 

Фиолетовый цвет снаружи картинки говорит о том, что все ее точки этого цвета будут при встраивании прозрачными. Встроенная копия будет выглядеть правильно на любом фоне:



2.7.4 Кадры

Картинка может состоять из нескольких кадров. Все кадры одной картинки имеют одинаковый размер, но различное содержимое. Кадры полезны для следующих целей:

- **Мультипликация.** Для картинки, встроенной в изображение, можно задать период смены кадров. В этом случае кадры будут автоматически сменять друг друга с заданным периодом;
- **Управляемая смена кадров.** Кадры встроенной картинки можно переключать в зависимости от состояния технологического процесса с помощью аниматора номера кадра. Это удобнее и эффективнее, чем подменять одну картинку другой, поскольку все кадры имеют один и тот же размер;
- **Группировка картинок.** Все картинки в Графите представлены одним линейным списком. Для удобства поиска вместо нескольких однотипных картинок можно сделать одну многокадровую картинку.

В окне редактирования картинки всегда виден только один из кадров. С помощью элементов меню **Картинка** можно добавлять и удалять кадры, а также переходить от одного кадра к другому.

2.7.5 Свойства картинок

Каждая картинка имеет имя, порядковый номер и срезы. Эти свойства имеют тот же смысл и редактируются тем же способом, как и для векторных изображений (2.5).

Кроме того, для картинки можно задать размер в диалоговом окне **Свойства картинки**, вызываемом из меню **Файл / Параметры окна**. Если при изменении размера картинки включить флажок **Растянуть изображение**, то картинка будет отмасштабирована пропорционально изменению размера. Если флажок не включать, картинка будет обрезана или дополнена прозрачным цветом.

Инв. № подл.	Подп. и дата
Подп. и дата	Инв. №
Взаим. инв. №	Подп. и дата
Инв. № подл.	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						20

3 Типы элементов

Как было сказано выше, все элементы делятся на **одиночные** и **групповые**. Одиночный элемент изображения – это такой элемент, внутреннее содержимое которого не может быть изменено в процессе редактирования данного изображения. Групповой элемент всегда состоит из других элементов и его содержимое доступно для редактирования.

Помимо **основных** типов элементов, всегда доступных в Графите, имеются **дополнительные** элементы, перечень которых разработчиками может быть в любой момент расширен. Дополнительные элементы всегда являются одиночными, поэтому в дальнейшем мы будем делить все типы элементов на 3 группы: основные, групповые и дополнительные (таблица 13).

Таблица 13 – Типы элементов

Основные	Отрезок	
	Прямоугольник	
	Эллипс	
	Сектор	
	Ломаная	
	Кривая	
	Текст	
	Шкала	
	Круговая шкала	
	Стрелка	
	Реквизит	
	Дата/время	
	Строка состояния	
	Отросток	
	Список состояний	
	Список изображений	
Вставка		
Дополнительные	График	
	Таблица данных из архива	
	Окно ошибок	
	Полис	
	Список параметров	
	Состояния архивных станций	
	Хроника дискретных событий	
	ActiveX	
Групповые	Объединение	
	Соединение	
	Серия	
	Кнопка	
	Уровень	
	Движок	
	Ползунок	

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

Описание каждого типа включает в себя следующие разделы (некоторые из них могут быть опущены):

- **Назначение** – описывает внешний вид элемента и его использование;
- **Рисование** – описывает правила добавления элемента в изображение;
- **Преобразования** – описывает, какие геометрические преобразования поддерживает элемент;
- **Настройка** – описывает контекстное меню и окна настройки элемента;
- **Точки соединения** – перечисляет точки соединения элемента;
- **Атрибуты** – перечисляет атрибуты элемента (кроме общих, таких как видимость, описанных в 2.6.8).


3.1 Основные

3.1.1 Отрезок

3.1.1.1 Назначение

Отображение прямой линии.

3.1.1.2 Рисование

Выберите инструмент добавления отрезка . Удерживая левую кнопку мыши, проведите линию желаемой длины, затем отпустите кнопку. Если нажата клавиша **Shift**, отрезок будет строго горизонтальным или вертикальным.

3.1.1.3 Преобразования

Элемент поддерживает масштабирование, сжатие, отражение, повороты и сдвигание. Инструмент изменения формы устанавливает для отрезка две контрольные точки – начало и конец. Переместив любую из них, можно изменить длину отрезка или расположение одной точки относительно другой.

3.1.1.4 Точки соединения

Точками соединения являются начало и конец отрезка.

3.1.1.5 Атрибуты


Для отрезка можно задать атрибуты контура и фона.

3.1.2 Прямоугольник

3.1.2.1 Назначение

Отображение прямоугольника, углы которого могут быть скруглены. Стороны прямоугольника строго параллельны сторонам экрана.

3.1.2.2 Рисование

Выберите инструмент добавления прямоугольника . Удерживая левую кнопку мыши, нарисуйте прямоугольник желаемого размера, затем отпустите кнопку. Если нажата клавиша **Shift**, добавляемый прямоугольник будет являться квадратом.

3.1.2.3 Преобразования

Элемент поддерживает масштабирование, сжатие, отражение и повороты на 90°.

Инструмент изменения формы устанавливает контрольные точки в вершинах прямоугольника. Изменение формы происходит между двумя крайними состояниями – от прямоугольника до эллипса. Перемещение контрольных точек возможно внутри границ (включая их) прямоугольника.

3.1.2.4 Точки соединения

Точками соединения являются середины сторон прямоугольника.

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						22

Если у многоугольника есть самопересечения, то образуются прозрачные области, не принадлежащие многоугольнику (рисунок 5 а).

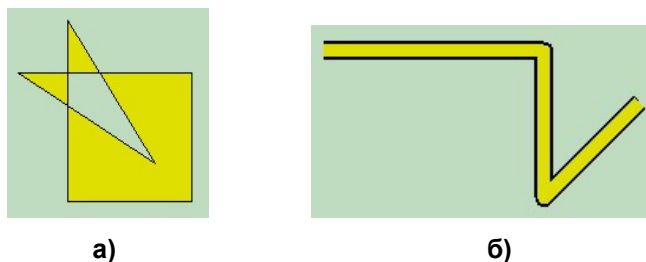


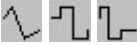
Рисунок 5 – Изображение ломаных

Трубчатые ломаные обычно используются для рисования трубопроводов. Для них отдельно задается цвет стенок «трубы» и цвет ее заполнения (рисунок 5 б). Все изломы трубы скругляются. Атрибут «толщина контура» для трубчатой ломаной означает толщину стенок трубы.

По способу задания все ломаные линии делятся на 2 вида: обычные и ступенчатые. Ступенчатые ломаные состоят из чередующихся вертикальных и горизонтальных ребер.

Начальная и конечная вершины ломаной обладают способностью «прилипнуть» к точкам соединения других элементов изображения.

3.1.5.2 Рисование

Выберите один из инструментов добавления ломаной . Первое ребро ломаной рисуется так же как обычный отрезок. Для отображения остальных ребер достаточно указать щелчком мыши следующую вершину ломаной.


Если ломаная должна быть **замкнутой**, то заключительное ребро тоже рисуется как обычный отрезок, линия которого ведется до пересечения с первой вершиной. Для того, чтобы замкнуть ломаную, наведите курсор на первую точку и, когда он примет форму мишени, отпустите левую кнопку мыши. Точно совместить конечную вершину с начальной довольно сложно, поэтому вводится допуск, величину которого можно задать в настройках.


При рисовании ступенчатых ломаных одно нажатие мыши добавляет сразу два ребра. Не стремитесь провести горизонтальную или вертикальную линию: в этом случае одно из добавленных ребер будет иметь нулевую длину, что может привести к путанице.

3.1.5.3 Преобразования

Элемент поддерживает масштабирование, сжатие, повороты на произвольный угол, в том числе на 90°, отражение и скашивание.

Инструмент изменения формы устанавливает контрольные точки в вершинах ломаной (для ступенчатых ломаных – только в четных вершинах). Перемещая контрольные точки в произвольных направлениях можно добиться желаемой конфигурации ломаной.

В режиме изменения формы можно также добавлять и удалять вершины ломаной. Для **удаления вершины** нажмите клавишу **Ctrl**, а затем нажмите левую кнопку мыши на той вершине, которую нужно удалить. У курсора изменения формы появляется значок . При отпускании кнопки мыши вершина будет удалена.

Чтобы **из одной вершины сделать две**, нажмите клавишу **Ctrl**, а затем нажмите левую кнопку мыши на нужной вершине и переместите курсор в другую точку. У курсора изменения формы появляется значок . При отпускании кнопки мыши будет добавлена вершина в точке отпускания.

3.1.5.4 Настройка

С помощью специального диалога можно изменить тип и вид ломаной, а для трубчатых ломаных – задать толщину стен трубки. Кроме этого, можно отредактировать численные значения координат вершин ломаной. В списке координат вершин можно добавлять или удалять строки. Подробнее о координатах см. 2.6.7.

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

3.1.5.5 Точки соединения

Точками соединения ломаной являются ее вершины. У замкнутой ломаной – все, у незамкнутой – кроме начальной и конечной, у ступенчатой – кроме двух начальных и двух конечных.

3.1.5.6 Атрибуты

Для незамкнутой ломаной можно задать атрибуты контура и фона, для замкнутой и трубчатой – атрибуты заполнения, контура и фона. Для трубчатой ломаной под контуром понимаются стенки трубы, под заполнением – внутренность трубы. Диаметр трубы задается отдельно. Чтобы избежать дефектов прорисовки, не рекомендуется для трубчатых ломаных задавать тонкий или пунктирный контур.

3.1.6 Кривая


3.1.6.1 Назначение

Отображение кривых линий различной конфигурации. Кривые могут быть незамкнутые и замкнутые, состоять из одного или нескольких звеньев.

Если у замкнутой кривой есть самопересечения, то образуются прозрачные области, не принадлежащие кривой (так же, как и у ломаных).

Начальная и конечная вершины кривой обладают способностью «прилипнуть» к точкам соединения других элементов изображения.

3.1.6.2 Рисование

Выберите инструмент добавления кривой . Удерживая левую кнопку мыши, нарисуйте кривую нужного размера и отпустите кнопку. Если требуется изобразить кривую более сложной формы, то операцию следует повторить. Каждая следующая кривая приклеивается к предыдущей, в результате чего получается одна многозвенная кривая.

3.1.6.3 Преобразования

Элемент поддерживает масштабирование, сжатие, повороты, отражение и скашивание.

Инструмент изменения формы устанавливает четыре контрольные точки для каждого звена кривой. Две из них находятся в начальной и конечной точке кривой, две другие определяют направление кривой в начальной и конечной точках.

Потянув за контрольную точку можно изменить конфигурацию кривой. Если кривая состоит из нескольких звеньев, то с помощью клавиши **Shift** можно изменить одновременно направление смежных звеньев.

3.1.6.4 Настройка

Выбрав пункт контекстного меню кривой **Изменить**, с помощью специального диалога можно указать тип кривой – замкнутый или незамкнутый.

3.1.6.5 Точки соединения




Точками соединения являются точки стыка смежных звеньев кривой.

3.1.6.6 Атрибуты

Для кривой можно задать атрибуты контура и фона, для замкнутой кривой – также атрибуты заполнения.

3.1.6.7 Пример рисования овала

Покажем на примере, как нарисовать овал яйцевидной формы.

- 1 Выберите инструмент  и нарисуйте одно звено кривой.
- 2 Чтобы не рисовались следующие звенья, выберите инструмент .
- 3 Щелкнув правой кнопкой мыши по кривой, укажите пункт меню **Изменить**.
- 4 В диалоговом окне настройки кривой укажите тип **Замкнутая**.
- 5 Выберите инструмент .

Инд. №	Подп. и дата
Инд. №	
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата

- 6 Удерживая клавишу **Shift**, нажмите левую кнопку мыши на одной из контрольных точек, не лежащих на кривой, и переместите эту точку симметрично относительно вершины кривой (рисунок 6 а).

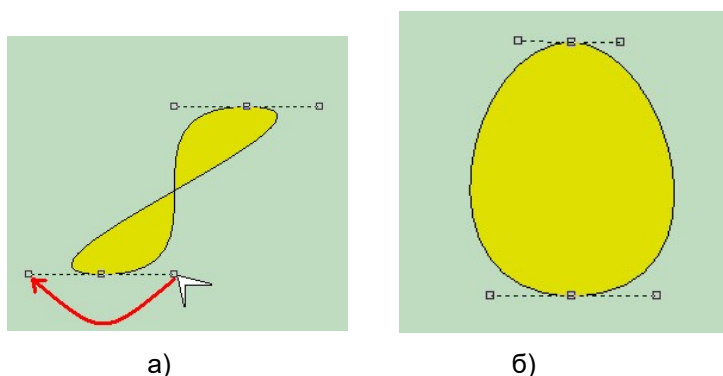


Рисунок 6 – Пример создания овала яйцевидной формы


- 7 Перемещая другие контрольные точки, придайте кривой нужную форму (рисунок 6 б).

3.1.7 Текст

3.1.7.1 Назначение

Отображение текста в прямоугольной области. Текст может располагаться по горизонтали или по вертикали, состоять из одной или нескольких строк. Прямоугольная область может иметь собственный цвет заполнения и рамки.

3.1.7.2 Рисование

Выберите инструмент добавления текста . Откроется диалоговое окно настройки текста, подробно описанное в 3.1.7.4. Наберите текст и нажмите кнопку **ОК**. Установите элемент на выбранное место.

Если при установке выполнить однократный щелчок левой кнопкой мыши, размер рамки добавляемого текста будет автоматически подогнан под введенный текст. Если же выделить мышью прямоугольник, добавляемый текст будет вписан в него.

3.1.7.3 Преобразования

Элемент поддерживает масштабирование, сжатие и повороты на 90°.

3.1.7.4 Настройка

Выбрав пункт контекстного меню текста **Изменить текст**, можно вызвать диалоговое окно настройки текста, которое содержит следующие элементы:

- 1 Поле ввода текста. В нем можно изменить содержание текста. При вводе многострочного текста можно использовать клавишу **Enter** для отделения строк. Поддерживаются стандартные операции копирования и вставки.
- 2 **Список стилей**. Можно выбрать требуемый стиль, а также отредактировать набор имеющихся стилей. Подробнее о стилях см. 6.1.2.
- 3 Флажок **Многострочный**.
- 4 Флажок **Перенос по словам**. Флажок действует только для многострочного текста. Если флажок включен, разбиение на строки происходит автоматически по пробелам; если выключен – то вручную, там где была нажата клавиша **Enter** при вводе текста.
- 5 Флажок **Автоматический размер**. Для однострочного текста флажок означает, что при любых дальнейших изменениях ширина и высота рамки будут автоматически подгоняться под размер текста. Для многострочного текста автоматически подгоняется только высота рамки.

Примечание - Функция автоматического размера работает корректно только для шрифтов типа **TrueType** или **OpenType**.

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

6 **Настройка табуляции.** Непосредственный ввод символов табуляции в диалоге не поддерживается, но данная настройка имеет смысл для текста, скопированного через буфер обмена.

3.1.7.5 Точки соединения

Точками соединения являются середины сторон прямоугольной области текста.

3.1.7.6 Атрибуты

Для элемента «**текст**» можно задать атрибуты текста, а также атрибуты контура и фона. Последние влияют на вид рамки, содержащей текст.

3.1.8 Шкала

3.1.8.1 Назначение




Отображение линейной шкалы. Шкала может быть направлена горизонтально или вертикально.

Изображение шкалы включает следующие элементы:

- **направляющая** (в виде отрезка);
- **риски** 3-х типов: крупные, средние и мелкие;
- **метки**, отображающие в виде текста значения на шкале, соответствующие крупным рискам.

Количество рисок и меток вычисляется автоматически в зависимости от размеров шкалы, шрифта меток и текущего масштаба изображения. Если размеры шкалы малы, мелкие и даже средние риски могут не выводиться.

3.1.8.2 Рисование

Выберите инструмент добавления шкалы . Установите элемент на выбранное место. Будет добавлена вертикальная шкала с диапазоном 0 – 100. Для получения горизонтальной шкалы используйте операцию поворота , для изменения ориентации рисок - операцию отражения .

3.1.8.3 Преобразования

Элемент поддерживает масштабирование, сжатие, отражение и повороты на 90°.

Инструмент изменения формы устанавливает для линейной шкалы 7 контрольных точек. Одна точка устанавливается на уровне меток, потянув за нее, можно регулировать расположение меток относительно шкалы. 3 пары точек устанавливаются на концах первых по порядку крупных, средних и мелких рисок; потянув за них, можно регулировать длину рисок каждого типа. Регулируя подобным образом длину рисок, можно из шкалы получить, к примеру, координатную сетку (рисунок 7).

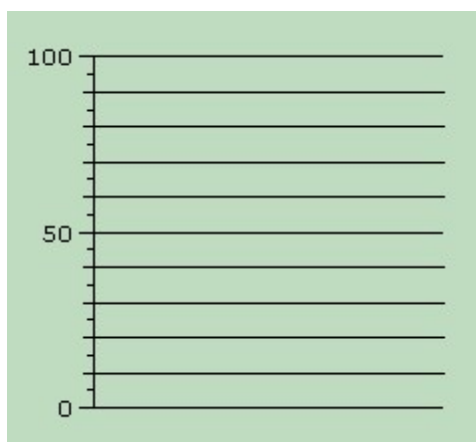


Рисунок 7 – Пример координатной сетки

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

3.1.8.4 Настройка

Для изменения диапазона шкалы вызовите пункт меню **Выбор параметра**. После выбора марки минимум и максимум шкалы будут соответствовать полям **Минимум** и **Максимум** закладки **Диапазоны** в форме **Аркады Объекты**.

3.1.8.5 Атрибуты



Для шкалы можно задать атрибуты контура (они определяют вид направляющей и рисунок шкалы), текста и фона (они определяют вид меток шкалы).

3.1.9 Круговая шкала

3.1.9.1 Назначение

Отображение круговой шкалы. Шкала располагается вдоль дуги окружности или эллипса.

3.1.9.2 Рисование

Выберите инструмент добавления шкал . В выпадающем меню выберите инструмент добавления круговой шкалы . Установите элемент на нужное место.

Если при установке выполнить однократный щелчок левой кнопкой мыши, будет создана шкала в виде половины окружности. Если же выделить мышью прямоугольник, будет создана шкала в виде половины эллипса, вписанного в этот прямоугольник.

3.1.9.3 Преобразования

Элемент поддерживает масштабирование, сжатие, отражение и повороты на 90° .

Инструмент изменения формы для круговой шкалы работает аналогично линейной шкале.

3.1.9.4 Настройка

Контекстное меню круговой шкалы содержит 3 специальных пункта:

- 1 **Изменить** – вызывает диалоговое окно настроек.
- 2 **Выбор параметра** – позволяет привязать диапазон шкалы к диапазону параметра. После выбора марки минимум и максимум шкалы будут соответствовать полям **Минимум** и **Максимум** закладки **Диапазоны** в форме **Аркады Объекты**.
- 3 **Инвертировать** – изменяет расположение рисок относительно дуги шкалы.

В диалоговом окне настроек можно указать 4 параметра (рисунок 8):

- 1 Расположение начальной и конечной точек дуги относительно центра в градусах:

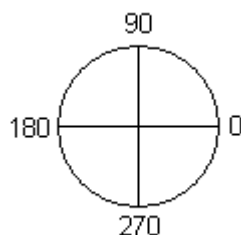


Рисунок 8 – Пример круговой шкалы

- 4 Направление шкалы – по часовой стрелке или против.
- 5 Видимость основной линии (т.е. дуги).
- 6 Видимость текстовых меток.

3.1.9.5 Атрибуты

Для круговой шкалы можно задать те же атрибуты, что и для линейной.



Инд. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата


3.1.10 Стрелка

3.1.10.1 Назначение

Отображение стрелки, служащей индикатором для круговой шкалы. Стрелка представляет собой отрезок, который может поворачиваться в зависимости от значения некоторого параметра.

3.1.10.2 Рисование

Выберите инструмент добавления шкал . В выпадающем меню выберите инструмент добавления стрелки . В диалоговом окне выберите параметр, который будет отвечать за положение стрелки. Установите элемент на нужное место.

Обычно требуется нарисовать круговую шкалу, совмещенную со стрелкой. Для этого в том же выпадающем меню нужно выбрать инструмент . При этом будет добавлено объединение двух элементов – круговой шкалы и стрелки – привязанных к одному и тому же параметру. В дальнейшем их можно редактировать по отдельности.

3.1.10.3 Преобразования

Элемент поддерживает масштабирование, сжатие, отражение и повороты на 90°.

Инструмент изменения формы устанавливает на концах стрелки две контрольные точки. Потянув за любую из них, можно изменить длину стрелки.

3.1.10.4 Настройка

Контекстное меню стрелки содержит два специальных пункта:

- **Изменить;**
- **Выбор параметра.**

Их назначение аналогично таким же пунктам для круговой шкалы.

3.1.10.5 Атрибуты

Для стрелки можно задать частоту мигания контура и шаблон контура. Цвет стрелки является инверсным по отношению к тем элементам, поверх которых стрелка нарисована.


3.1.11 Реквизит

3.1.11.1 Назначение

Реквизит – это статическая текстовая информация об объекте или его параметре. Слово «статическая» означает, что значение реквизита вводится на этапе проектирования АСУТП и не зависит от состояния объекта в оперативном режиме. Значения реквизитов хранятся в БД проекта и вводятся в других приложениях САПР, главным образом, в **Аркаде**. Примером реквизита может служить марка объекта.

Графит позволяет создавать типовые изображения объектов. Предположим, в объектном окне задвиги оператор должен видеть марку задвиги. При рисовании объектного окна нельзя добавить элемент типа «**текст**» с конкретным значением марки, потому что одно и то же изображение будет использоваться для объектов с разными марками. В этой ситуации следует использовать элемент типа «**реквизит**».

3.1.11.2 Рисование

Выберите инструмент добавления реквизита . В диалоговом окне осуществите настройку реквизита. Установите элемент на нужное место.

3.1.11.3 Преобразования

Элемент поддерживает масштабирование, сжатие и повороты на 90°.

3.1.11.4 Настройка

Окно настройки реквизита вызывается с помощью элемента контекстного меню **Изменить реквизит**. В окне указывается тип реквизита (таблица 14) и, если необходимо, дополнительные настройки.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. №	Подп. и дата
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						29

Таблица 14 – Типы реквизитов

Тип реквизита	Где вводится	Доп. настройки
Марка	Аркада – Объекты – Основная таблица – Поле «Марка»	
Имя объекта	Аркада – Объекты – Основная таблица – Поле «Полное имя»	
Обозначение объекта	Аркада – Объекты – Основная таблица – Поле «Обозначение»	
Диспетчерское наименование	Аркада – Объекты – Основная таблица – Поле «Дисп. наименование»	
Номер объекта	Аркада – Объекты – Основная таблица – Поле «Номер»	
Адрес объекта	Аркада – Объекты – Основная таблица – Поле «Адрес»	
Единица измерения параметра	Аркада – Объекты – Диапазоны – Поле «Размерность»	Параметр объекта
Оттекстовка значения параметра	Аркада – Объекты – Оттекстовка	Параметр объекта, значение параметра
Значение доп. Атрибута	Аркада – Объекты – Атрибуты	Название атрибута

3.1.11.5 Точки соединения

Те же, что у элемента **Текст** (см. 3.1.7).

3.1.11.6 Атрибуты


Те же, что у элемента **Текст** (см. 3.1.7).

3.1.12 Дата-время

3.1.12.1 Назначение

Отображение календарной даты или времени. Обычно, отображаются текущие дата и время по часам операторской станции. В режиме проигрывания архива отображаются дата и время, установленные в проигрывателе.

3.1.12.2 Рисование

Выберите инструмент добавления даты-времени . Выберите тип элемента – дата или время. Установите элемент на выбранное место.

3.1.12.3 Преобразования

Элемент поддерживает масштабирование, сжатие и повороты на 90°.

3.1.12.4 Настройка

С помощью специального диалога можно задать тип элемента – дата или время.

3.1.12.5 Точки соединения

Те же, что у элемента **Текст** (см. 3.1.7).

3.1.12.6 Атрибуты

Те же, что у элемента **Текст** (см. 3.1.7).

3.1.13 Строка состояния

3.1.13.1 Назначение

Отображение текстовой информации общего характера, не связанной с конкретным объектом.

3.1.13.2 Рисование

Выберите инструмент добавления строки состояния . Осуществите настройку элемента (см. ниже). Установите элемент на нужное место.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						30

3.1.13.3 Преобразования

Элемент поддерживает масштабирование, сжатие и повороты на 90°.

3.1.13.4 Настройка

Выбрав пункт контекстного меню «Изменить», с помощью специального диалога можно задать тип выводимой в строке состояния информации (таблица 15):

Таблица 15 – Типы информации

Технологические ошибки	Текст последней технологической ошибки
Приборные ошибки	Текст последней приборной ошибки
Системные ошибки	Текст последней системной ошибки
Ошибки оператора	Текст сообщения, формируемого в ответ на некорректные действия оператора (например, неправильно введенный номер объекта)
Прочие ошибки	В настоящее время не используется
Состояние системы	В настоящее время не используется
Подсказки	Информация о мнемосимволе, над которым находится курсор, а также информация о действиях, которые будут произведены при щелчке мышью в данной точке экрана.

3.1.13.5 Точки соединения

Те же, что у элемента **Текст** (см. 3.1.7).

3.1.13.6 Атрибуты

Те же, что у элемента **Текст** (см. 3.1.7).

3.1.14 Отросток

3.1.14.1 Назначение

Отросток является дополнительной точкой соединения, если он входит в состав группового элемента или вставленного изображения. Отростки нужны только в редакторе, в оперативном режиме они невидимы.

Отросток задает не только позицию, но и желательное направление, в котором должно выходить соединение. Отросток изображается в виде буквы «Т», ножка которой задает направление, а середина перекладины – позицию требуемой точки соединения.

3.1.14.2 Рисование

Выберите инструмент добавления отростка . Установите элемент на выбранное место.

3.1.14.3 Преобразования

Элемент поддерживает отражения и повороты на любой угол. Его можно также повернуть с помощью инструмента изменения формы.

Например, чтобы к мнемосимволу задвижки  можно было бы присоединить изображение трубы, следует при проектировании мнемосимвола добавить два отростка на торцах задвижки:



3.1.15 Список состояний


3.1.15.1 Назначение

Показывает список всех возможных состояний указанного параметра. Строка списка, соответствующая текущему состоянию параметра, выделяется цветом. Состояния должны быть поименованы в **Аркаде** (форма **Объекты**, закладка **Отекстовка**), либо в Пилоне (в редакторе логических программ).

Инд. №	Подп. и дата
Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Подп. и дата

Список состояний можно также использовать для параметра типа «Набор бит» (например, Условия логической программы). В этом случае список содержит все поименованные биты значения, и цвет строки определяется значением соответствующего бита.

3.1.15.2 Рисование

Выберите инструмент добавления списка состояний . Откроется диалоговое окно **Выбор марки и параметра**. Выберите параметр, список состояний которого нужно отобразить. Осуществите настройку элемента (см. ниже). Установите элемент на нужное место.

3.1.15.3 Преобразования

Элемент поддерживает масштабирование и сжатие.

3.1.15.4 Настройка

Контекстное меню списка состояний содержит 2 специальных пункта:

- 1 **Выбор параметра** - Вызывает диалоговое окно **Выбор марки и параметра** и позволяет выбрать параметр, список состояний которого нужно отобразить;
- 2 **Настройка списка** - Вызывает диалоговое окно настройки списка состояний. В зависимости от типа параметра, окно настройки будет иметь разный вид. Для параметра типа «Состояние» в окне можно задать тип списка – простой или раскрывающийся; если установить флажок **Допускается управление**, то в оперативном режиме значение параметра можно будет изменить, выбрав мышью требуемую строку списка. Для параметра типа «Набор бит» в окне можно указать, следует выделять строки со значением бита 1 или 0. Можно также выводить не все строки, а только выделенные.

3.1.15.5 Атрибуты

Для списка состояний можно задать следующие атрибуты:

- цвет и частота мигания заполнения – определяют вид выделенных строк;
- цвет и частота мигания фона – определяют вид невыделенных строк;
- цвет, частота мигания и шаблон контура – определяют вид рамки;
- цвет, частота мигания и стиль текста – определяют параметры текста.


3.1.16 Список изображений

3.1.16.1 Назначение

Показывает список всех объектов определенного типа, удовлетворяющих заданным условиям. Каждый объект изображается в виде заранее созданного мнемосимвола. Мнемосимволы располагаются в одну или несколько колонок или строк. При необходимости на списке появляются полосы прокрутки.

Каждый мнемосимвол - элемент списка является «живым», т.е. может содержать в себе элементы анимации и управления.

3.1.16.2 Рисование

Выберите инструмент добавления списка изображений . Сначала откроется диалоговое окно выбора мнемосимвола. Выберите нужный тип объекта и мнемосимвол, который будет изображать объекты в списке. После нажатия кнопки **ОК** откроется диалоговое окно настроек списка, описанное ниже в пункте «**Настройка**». Если в этом окне ничего не заполнять, то список будет содержать все объекты указанного типа. Нажмите кнопку **ОК** и установите элемент на нужное место.

3.1.16.3 Преобразования

Элемент поддерживает масштабирование и сжатие.

3.1.16.4 Настройка

Список изображений можно настроить, чтобы он выводил не все объекты указанного типа, а только те, которые удовлетворяют определенным условиям - фильтрам. Условие записывается на языке встроенной программы. Этот язык описан в 5.4. Итак, в настройках списка указывается:

Изн. № подл.	Подп. и дата	Взаим. инв. №	Изн. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

- **Статический фильтр.** Представляет собой выражение числового типа. Это выражение вычисляется один раз для каждого объекта при открытии мнемокадра, содержащего список. Объекты, для которых значение выражения равно 0, отбрасываются, для оставшихся объектов запускается опрос параметров из контроллера. Выражение статического фильтра должно использовать только статические свойства объектов, т.е. свойства, хранящиеся в БД (например, **Марка**). Если выражение не задано, через статический фильтр проходят все объекты данного типа;
- **Динамический фильтр.** Представляет собой выражение числового типа. Выражение периодически вычисляется для каждого объекта, прошедшего статический фильтр. Объект отображается в списке, если значение выражения не равно 0. Выражение динамического фильтра может содержать значения параметров объектов, получаемых из контроллера. Если выражение не задано, отображаются все объекты, прошедшие статический фильтр;
- **Параметр сортировки.** Представляет собой выражение числового или строкового типа. Выводимые объекты будут отсортированы в порядке возрастания возвращаемого значения этого выражения. Тип возвращаемого значения должен быть одинаковым для всех объектов. Если выражение не задано, объекты не сортируются.

3.1.16.5 Атрибуты

Для списка изображений атрибут «цвет заполнения» определяет фон, на котором будут изображаться мнемосимволы.

3.1.16.6 Примеры

Рассмотрим два примера:

- 1 Вывести все аналоговые датчики из узла "Котел", находящиеся в подмене. Отсортировать по марке.

Статический фильтр: `Band.BandName = 'Котел'`

Динамический фильтр: `Режим = Режим.Подмена`

Параметр сортировки: `Марка`

- 2 Вывести все датчики давления, генерирующие аварийную ошибку. Отсортировать по величине давления.

Статический фильтр: `Card.ТecTypeId = 16843008`

Динамический фильтр: `(ТехнАварНовая = ТехнАварНовая.Есть) and (ТехнАварПредупр = ТехнАварПредупр.Есть)`

Параметр сортировки: `ЗначениеСигнала`

3.1.17 Вставка

3.1.17.1 Назначение

Самый мощный инструмент Графита с точки зрения экономии труда проектанта – это вставка одних изображений в другие. Типовые фрагменты мнемосхем не нужно рисовать много раз: достаточно сделать одно изображение этого фрагмента и затем вставлять его везде, где требуется.

Вставленное изображение кратко называется **вставкой**. Вставку можно масштабировать, поворачивать, словом, делать все те геометрические преобразования, которые допускают элементы вставленного изображения. При этом исходное изображение не изменяется. Если же отредактировать исходное изображение, то изменится его образ во всех местах, куда оно было вставлено.


Вставка включает в себя не только образ исходного изображения, но также всю его анимацию и управление. При вставке типового изображения (мнемосимвола или объектного окна) в мнемосхему необходимо указать для вставки марку; после этого вставка будет отображать состояние конкретного объекта.

Вставка представляет собой одиночный элемент изображения, т.е. внутреннее ее содержимое недоступно. Тем не менее, внешний вид вставки можно в определенных пределах изменить. Часто

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

типовые фрагменты мнемосхем отличаются лишь цветом, скажем, задвижки на газе изображают голубым цветом, а задвижки на воде – зеленым. Для этого не нужно изготавливать два изображения задвижки. Достаточно при рисовании исходного изображения задвижки указать, что требуемые элементы задвижки имеют **внешний цвет**. Впоследствии цвет этих элементов на вставке можно изменить, определив для вставки цвет заполнения (щелчком левой кнопки по палитре).

3.1.17.2 Рисование

Выберите инструмент вставки  и в выпадающем меню укажите тип изображения. В открывшемся диалоговом окне выберите изображение, которое нужно вставить (3.1.17.4). Установите изображение на нужное место.

Если при установке выполнить однократный щелчок левой кнопкой мыши, то вставленный элемент будет иметь тот же размер, что и исходное изображение. Если же выделить мышью прямоугольник, добавляемый элемент будет вписан в него.

3.1.17.3 Преобразования

Каждому объекту в **Аркаде** можно присвоить **номер**. Этот номер может использоваться оператором для быстрого поиска объекта с помощью клавиатуры. Поле номера автоматически изображается рядом со вставкой в случае, если последняя связана с маркой. Номер всегда изображается горизонтально, даже если повернуть вставку. Инструмент изменения формы позволяет перемещать поле номера относительно вставки.

3.1.17.4 Настройка

Контекстное меню вставки содержит следующие специальные пункты:

- **Привязка** – вызывает диалог привязки;
- **Редактировать исходное изображение** – открывает исходное изображение вставки в отдельном окне для редактирования;
- **Исходный размер** – восстанавливает размер и ориентацию вставки, как у исходного изображения. Все прочие свойства вставки не изменяются;
- **Аркада** – если вставка связана с маркой, запускает Аркаду и в форме **Объекты** устанавливает указатель на эту марку;
- **Пилон** – если вставка связана с маркой и марка привязана к алгоблоку, то открывает в Пилоне задачу с этим алгоблоком.

Диалоговое окно привязки зависит от типа вставленного изображения и от типа редактируемого изображения. В любом случае, в диалоге можно выбрать вставляемое изображение из списка, обладающего следующими свойствами:

- если вставляется мнемосхема, этот список сгруппирован по узлам;
- если типовое изображение вставляется в мнемосхему, список сгруппирован по оперативному типу. Дополнительно требуется указать марку, с которой будет связана вставка;
- если одно типовое изображение вставляется в другое, причем того же типа, достаточно выбрать изображение из списка;
- в типовое изображение составного типа можно вставить типовое изображение любого его члена – в этом случае диалог содержит дерево членов и список соответствующих изображений;
- при вставке картинки можно также выбрать нужный номер кадра картинки и период смены кадров в оперативном режиме. Номер кадра и период смены кадров можно также изменять с помощью аниматоров.

3.1.17.5 Атрибуты

Для вставленных **векторных** изображений (мнемосимволов и др.) определены следующие атрибуты:

- **Цвет заполнения** – определяет цвет тех элементов мнемосимвола, которые были нарисованы «внешним» цветом;
- **Цвет текста и цвет фона** – определяют вид поля номера. Чтобы подавить вывод номера, следует установить цвет текста прозрачным.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

3.2.2.4 Настройка

Диалоговое окно настройки содержит флажки:

- **Всегда перерисовывать полностью.** - Если флажок выключен, перерисовываются только те элементы объединения, атрибуты которых были изменены аниматорами. Если флажок включен, объединение всегда перерисовывается целиком. Включайте флажок, если мигающие или анимированные элементы объединения частично перекрываются другими элементами. Это предотвратит затирание других элементов. Аналогичной цели служит аниматор перерисовки, но его сложнее настраивать;
- **Буферизованный вывод.** - Если флажок включен, то элементы объединения вначале рисуются на невидимом «заднем буфере», а потом этот буфер копируется на экран. Использование этого флажка предотвращает смагивание элементов объединения при перерисовке.

Следует учитывать, что каждый из этих флажков увеличивает время перерисовки изображения.

3.2.3 Соединение

3.2.3.1 Назначение

Соединение нескольких элементов в один с образованием в местах пересечения прозрачных областей, не принадлежащих получившемуся элементу (например, круг с прямоугольным отверстием). Отдельные элементы теряют свой индивидуальный цвет и принимают общий цвет соединения. Соединение имеет смысл только для геометрических элементов: прямоугольники, круги, многоугольники и т.д.

3.2.3.2 Рисование

Выделите элементы, которые необходимо соединить. Нажмите кнопку . Элементы примут цвет, установленный по умолчанию.

3.2.3.3 Преобразования

Элементы соединения допускают те геометрические преобразования, которые допускает каждый элемент группы

3.2.3.4 Атрибуты

Задание атрибутов для соединения не подчиняется общему принципу задания атрибутов групповых элементов. Все элементы соединения рисуются с атрибутами заполнения, контура и фона, заданными для соединения.

3.2.4 Серия



3.2.4.1 Назначение

Серия изображает некоторый фрагмент несколько раз со смещением вдоль прямой (рисунок 9). Серию удобно использовать для создания пунктирных линий, орнаментов, сеток и др.



Рисунок 9 – Пример использования серии

3.2.4.2 Рисование

Выделите элемент или несколько элементов, которые нужно размножить. Нажмите кнопку . В появившемся диалоговом окне задайте количество повторов. Шаг смещения серии будет направлен вправо. Для его коррекции используйте инструмент .

3.2.4.3 Преобразования

Серия элементов допускает те геометрические преобразования, которые допускает каждый элемент. Инструмент изменения формы позволяет задать длину и направление шага смещения

Инд. №	Подп. и дата
Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Подп. и дата

для серии. При этом, в отличие от операции масштабирования, не меняется размер входящих в серию элементов.

3.2.4.4 Настройка

Выбрав пункт контекстного меню **Изменить**, с помощью специального диалога можно задать количество повторов в серии.

3.2.4.5 Атрибуты

Подчиняются общему принципу задания атрибутов групповых элементов.


3.2.5 Кнопка

3.2.5.1 Назначение

Экранная кнопка, которая нажимается левой кнопкой мыши. На поверхности кнопки могут находиться любые элементы: текст, картинка и т.д. Эти элементы могут быть анимированными.

Кнопка не нажимается, если для нее установлен запрет управления.

3.2.5.2 Рисование

Нарисуйте и выделите элементы, которые должна содержать кнопка. Выполните команду создания кнопки . Размер кнопки будет автоматически подобран так, чтобы включить все выделенные элементы.

3.2.5.3 Преобразования

Кнопка допускает те геометрические преобразования, которые допускает каждый элемент кнопки, за исключением поворотов на произвольный угол и сжатия.

3.2.5.4 Атрибуты

Для кнопки можно задать следующие атрибуты:

- **Цвет и частота мигания заполнения** – определяет вид лицевой поверхности кнопки;
- **Цвет и частота мигания контура** - определяет вид прямоугольной рамки кнопки.

Остальные атрибуты подчиняются общим правилам задания атрибутов групповых элементов.

3.2.6 Уровень


3.2.6.1 Назначение

Выполняет закрашивание другим цветом некоторой части элемента (рисунок 10). Высота закрашенной части в оперативном режиме будет пропорциональна текущему значению указанного параметра. По умолчанию, минимальному значению параметра соответствует отсутствие закрашки, а максимальному – закрашки «доверху». На рисунке уровень включает в себя прямоугольник и текст и закрашен на 45%:




Рисунок 10 – Пример использования уровня

Линия уровня может быть как горизонтальной, так и вертикальной (но не наклонной).

Чаще всего уровень используется в качестве столбикового индикатора, в этом случае он создается на основе прямоугольника. Для быстрого создания столбика имеется специальная кнопка .

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Подп. и дата

3.2.6.2 Рисование

Нарисуйте и выделите элементы, которые должны закрашиваться. Нажмите кнопку . В диалоговом окне **Выбор параметра и марки** выберите параметр, уровень которого должен быть отображен. Выделенные элементы станут принадлежностью элемента «Уровень». К этому элементу будет добавлен аниматор шкалы. Аниматор можно увидеть и настроить в списке аниматоров для уровня. Можно также удалить этот аниматор и изменить высоту уровня с помощью встроенной программы (пример: ШкалЗнач := 0.45).

3.2.6.3 Преобразования

Уровень допускает те геометрические преобразования, которые допускает каждый элемент уровня, за исключением поворотов на произвольный угол и сжатия.

Инструмент изменения устанавливает 2 контрольные точки. Через одну из них будет проходить уровень при минимальном значении параметра, через другую – при максимальном. Если расстояние между точками по вертикали больше, чем по горизонтали, то линия уровня будет горизонтальной, иначе – вертикальной.

3.2.6.4 Настройка

Выбрав пункт контекстного меню **Изменить**, с помощью специального диалога можно задать цвет закрашиваемой части, а также что именно нужно закрашивать: заполнение, контур, фон или текст. По умолчанию, уровень изменяет цвет заполнения. На рисунке 11 изображен тот же уровень, что и выше, но для него установлено закрашивание текста:



Рисунок 11 – Пример использования закрашивания текста

3.2.6.5 Атрибуты

Для уровня всегда можно задать цвет заполнения. Этот атрибут определяет цвет закрашиваемой части уровня. Атрибут «Уровень шкалы» определяет процент закрашенной части. Остальные атрибуты подчиняются общим правилам задания атрибутов групповых элементов.

3.2.7 Движок

3.2.7.1 Назначение

Движок в оперативном режиме изменяет положение одного или нескольких элементов пропорционально значению некоторого параметра. Если значение параметра равно «**минимуму**», то положение элемента будет таким, как оно задано в редакторе. Если значение параметра равно «**максимуму**», то элемент переместится в положение, отмеченное пунктирной стрелкой – **направляющей** движка. (В режиме выполнения направляющая не отображается.)

С помощью движка можно не только изменять положение элемента, но и его размер.

Движок только отображает значение параметра, но не позволяет управлять параметром. Для управления параметром используйте элемент **Ползунок**.

3.2.7.2 Рисование

Нарисуйте и выделите элементы, которые должны перемещаться. Нажмите кнопку **Движок** . В открывшемся диалоговом окне **Выбор марки и параметра** выберите марку и параметр, который должен отвечать за положение движка. Будет создан движок с направляющей, направленной вверх. К движку будет автоматически добавлен аниматор шкалы. Аниматор можно увидеть и настроить в списке аниматоров для движка. Можно также удалить этот аниматор и управлять движком с помощью встроенной программы (пример: ШкалЗнач := 0.45).

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

3.2.7.3 Преобразования

Движок допускает те геометрические преобразования, которые допускает каждый элемент движка, за исключением поворотов на произвольный угол и скашивания.

Режим изменения формы устанавливает контрольную точку на конце направляющей движка. Потянув за нее, можно изменить направление и длину направляющей.

3.2.7.4 Настройка

Контекстное меню движка содержит пункт **Отображение**. Выбрав его, с помощью специального диалога можно задать параметры работы движка:

- сдвиги по осям X и Y – координаты вектора направляющей. Подробнее о координатах - см. подраздел 2.6.7;
- растяжение по осям X и Y – задают изменение размеров элемента. Например, если задать растяжение по оси X = 1, а растяжение по оси Y = 0.5, то при максимальном значении параметра элемент сожмется по вертикали в 2 раза;
- соответствие исходной позиции минимуму или максимуму параметра. Под исходной позицией понимается положение элемента, заданное в редакторе. По умолчанию, элемент в режиме выполнения занимает исходную позицию, когда значение параметра минимально. С помощью этой настройки можно обратить перемещение движка. Это оказывается необходимым в следующей ситуации. Предположим, при минимальном значении параметра высота элемента должна быть равна 0, а при увеличении параметра высота должна увеличиваться. В этом случае следует изобразить элемент так, как он должен выглядеть при максимальном значении параметра, а затем в настройках указать: растяжение по X = 1, растяжение по Y = 0 и соответствие исходной позиции максимуму параметра.

3.2.7.5 Атрибуты

Атрибуты подчиняются общим правилам задания атрибутов групповых элементов. Дополнительно определен атрибут **«Уровень шкалы»**, определяющий процент перемещения вдоль направляющей.


3.2.8 Ползунок

3.2.8.1 Назначение

Ползунок позволяет оператору изменять значение параметра путем перемещения некоторого элемента изображения вдоль прямой. Оператор должен нажать левую кнопку мыши в поле ползунка и затем перемещать ползунок в требуемом направлении, не отпуская кнопку мыши. В процессе перемещения значение параметра не изменяется – меняется только предустановка. При отпускании кнопки мыши значение параметра будет изменено. Чтобы изменение не произошло, достаточно увести курсор мыши из области ползунка (в перпендикулярном направлении).

С точки зрения редактирования ползунок аналогичен элементу **«Движок»**, за исключением того, что связь ползунка с параметром реализована внутренним способом, а не через аниматор. Ползунок всегда связан с предустановкой параметра.

3.2.8.2 Рисование

Нарисуйте и выделите элементы, которые должны выполнять функцию ползунка. Нажмите кнопку . В открывшемся диалоговом окне **Выбор марки и параметра** выберите марку и параметр, которым должен управлять ползунок. Будет создан ползунок с направляющей, направленной вверх.

3.2.8.3 Преобразования

Ползунок допускает те геометрические преобразования, которые допускает каждый элемент ползунка, за исключением поворотов на произвольный угол и скашивания.

Режим изменения формы устанавливает контрольную точку на конце направляющей ползунка. Потянув за нее, можно изменить направление и длину направляющей.

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

3.3.2.2 Особенности использования в Графите

График может представляться в двух вариантах:

- как элемент мнемосхемы;
- как элемент объектного окна .

В первом случае на одном графике могут сочетаться любые параметры любых объектов, отраженных в базе данных. Каждый график может содержать любое число кривых, - ограничение связано лишь с удобством рассмотрения этих кривых при оперативной работе. На одной мнемосхеме с графиком могут быть представлены любые другие изображения – технологические схемы, мнемосимволы, другие графики и т.д. Как и любая другая мнемосхема, каждая мнемосхема с графиком подготавливается индивидуально и вызывается по стандартной процедуре вызова мнемосхем. Такая мнемосхема может иметь переменный или фиксированный размер и открываться в нормальном, развернутом или свернутом виде.

Во втором случае на графике могут представляться любые сочетания любых параметров, но только одного выбранного объекта. Такие графики подготавливаются для каждого оперативного типа объекта (например, для объектов типа «Аналоговый датчик», «Регулятор» и т.п.). В оперативном режиме окно с графиком вызывается через мнемосимвол или любое другое изображение, к которому привязан рецептор вызова такого окна.

Пока изображение находится в режиме редактирования, все внутренние элементы графика неактивны. Для редактирования графика следует нажать правую кнопку мыши и из контекстного меню элемента выбрать пункт **Изменить**.

3.3.2.3 Особенности работы в режиме выполнения

В режиме выполнения Графита и операторской станции в графике доступно выпадающее меню, вызываемое кнопкой **Свойства** (рисунок 12).

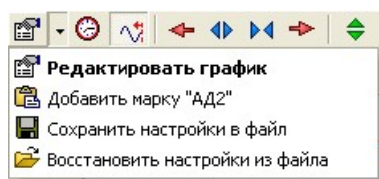


Рисунок 12 – Меню Свойства

С помощью этого меню с графиком можно производить следующие действия:

- сохранять и восстанавливать в файл формата Станции анализа (*.tcr), при этом в файл сохраняются текущие настройки графика, в том числе и те, которые сделаны «на ходу» (т.е. без сохранения в БД);
- автоматически вставить в график кривую для марки, сохраненной в буфере обмена Windows. В буфер обмена марку можно поместить обычным способом (выделив имя марки в другом приложении и выполнив системную операцию «Копировать») или с помощью рецептора «Запомнить марку». Если находящийся в буфере обмена текст не является маркой, находящейся в текущей БД, то этот пункт меню будет неактивным. При добавлении кривой будет выбран главный параметр данного оперативного типа (главный параметр для каждого оперативного типа задается разработчиком Квинта).

3.3.3 Таблица данных из архива

3.3.3.1 Назначение


Данный элемент позволяет просматривать любую архивную информацию в табличной форме. Информация всегда представляется в хронологической последовательности.

Графит использует ту же компоненту таблиц, что и в станция анализа. Подробное описание работы с таблицами приведено в руководстве по станции анализа.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

3.3.3.2 Особенности использования в Графите

3.3.3.2.1 Режим работы

Кнопка  и пункт меню **Режим работы** доступны только в режиме редактирования. В режиме обзора, а также в операторской станции они недоступны. Соответственно, настройки, входящие в диалог "**Режим работы**" можно изменять только при проектировании мнемосхемы. Эти настройки следующие:

- разрешать изменять: состав параметров, видимость параметров, внешний вид;
- просмотр только текущих данных;
- отображать данные;
- тема (вид настройки атрибутов текста - цвета и начертания для различных сообщений).

3.3.3.2.2 Типизация

В типовых изображениях используются типизированные таблицы с тем типом объекта, как у изображения. В нетиповых изображениях используются общие таблицы.

В типизированной таблице выбор параметра происходит в контексте какого-либо объекта (марки) заданного типа, но сам объект еще не задан. В режиме же обзора эта марка уже известна, и могут быть получены данные по заданным параметрам. Когда параметр добавляется или изменяется в режиме обзора, при известной марке, то не изображаются параметры объекта, которых нет в архиве, в отличие от изменений в режиме редактирования.

3.3.3.2.3 Пример использования

В 4.5 показано, как с помощью таблицы данных сформировать список технологических событий, Событийной станции.

3.3.4 Окно ошибок

3.3.4.1 Назначение

Окно ошибок отображает список актуальных на данный момент ошибок (технологических, приборных и др.). В ранних версиях Квинта список ошибок можно вызывать с помощью рецептора, как плавающее окно. В Квинте имеется дополнительная возможность: точно такой же список можно сделать элементом любого изображения. Такие встроенные списки имеют более тонкую настройку и позволяют реализовать функции **Приборной станции**.

3.3.4.2 Понятие ошибки

Под ошибкой в данном контексте понимается информация об отказе или неправильной работе какого-либо элемента оборудования или АСУ ТП. Каждая ошибка имеет некоторую продолжительность во времени – от момента появления до момента исчезновения. Ошибка называется **актуальной**, если она уже появилась, но еще не исчезла.

Актуальная ошибка может быть квитирована оператором. Квитируя ошибку, оператор как бы расписывается в том, что он принял эту ошибку к сведению. Пока актуальная ошибка никем не квитирована, она называется **новой**, после этого – **квитированной**.

В окне ошибок могут отображаться не только актуальные ошибки, но и старые. **Старой** называется ошибка, которая появилась, исчезла, но никем не была квитирована за это время. При квитировании старая ошибка удаляется из списка.

Чтобы старые ошибки показывались бы в списке, нужно в приложении **Настройка – Параметры** указать опцию **Сигнализация – Хранить старые ошибки** и задать время их хранения.

3.3.4.3 Механизм приема ошибок

Механизм получения ошибок Операторской станцией полезно представлять для лучшего понимания особенностей работы окна ошибок.

Каждый абонент-источник ошибок (контроллер, шлюз, архивная станция и др.) выдает список своих актуальных ошибок раз в 10 секунд. Это сообщение получают все «заинтересованные» абоненты, в нашем случае, операторская станция. Каждая ошибка в сообщении снабжена признаком, указывающим, была ли она квитирована.

Инд. №	Инд. №	Взаим. инв. №	Подп. и дата	Подп. и дата	ПФДИ.421457.009 И3.5	Лист
						42
Изм	Лист	№ докум	Подп.	Дата		

Операторская станция хранит (в ОЗУ) буфер актуальных (и, если требуется, старых) ошибок от всех абонентов. Для каждой ошибки в буфере хранится время появления, время последнего получения и время квитирования. Получив сообщение с ошибками от абонента-источника, операторская станция разбирает входящие в него ошибки и для каждой ошибки делает следующее:

- 1 Если ошибки нет в буфере – добавляет ее и инициализирует время появления и последнего получения ошибки.
- 2 Если ошибка есть в буфере и она старая – помечает ее как актуальную и обновляет время появления и последнего получения ошибки.
- 3 Если ошибка есть в буфере и она актуальная – обновляет время последнего получения ошибки.
- 4 Помечает как старые все ошибки в буфере, связанные с этим абонентом-источником, которых не было в пришедшем сообщении.

Кроме того, Операторская станция регулярно, по таймеру, просматривает свой буфер ошибок и для каждой ошибки выполняет следующее:

- 5 Если ошибка новая и не приходила за последние 15 секунд – помечает ее как старую и инициализирует время квитирования.
- 6 Если ошибка квитированная и не приходила за последние 15 секунд – удаляет ее из буфера.
- 7 Если ошибка старая и исчезла более 12 часов назад – удаляет ее из буфера.

Примечание - Время удержания новых ошибок и время хранения старых ошибок настраивается в приложении **Настройка / Параметры**.


3.3.4.4 Работа с окном в оперативном режиме

Окно ошибок содержит список ошибок и панель инструментов (панель может быть скрыта).

В списке для каждой ошибки отображается:

- **Время появления.** Более точно это время, когда Операторская станция первый раз получила сообщение, содержащее эту ошибку. Поэтому:
 - а) время фиксируется с точностью 10 мсек;
 - б) если Операторская станция перезапускалась после возникновения ошибки, то это будет время последнего перезапуска.
- **Время квитирования.** Для новых ошибок эта ячейка пуста. Для квитированных ошибок - это время, когда ошибка была квитирована. Если Операторская станция перезапускалась после квитирования этой ошибки, то в графе **Время квитирования** пишется «квит». Для старых ошибок - это время исчезновения;
- **Описание ошибки.** Текст, описывающий, **что** произошло;
- **Описание источника.** Текст, описывающий, **где** произошла ошибка.

Панель инструментов содержит следующие элементы:

- кнопка **Квитир..** Квитирует выбранные ошибки (в списке можно выбрать несколько ошибок, используя клавиши **Ctrl** или **Shift**);
- кнопка **Мнемо.** Вызывает мнемосхему с мнемосимволом объекта, с которым связана выбранная ошибка. Мнемосимвол выделяется рамкой и автоматически открывается оперативное окно для этого объекта. Если мнемосимвол объекта установлен на нескольких мнемосхемах и открывается не та, которая нужна, проектанту следует запустить **Аркаду**, открыть форму **Объекты**, найти данный объект и выбрать закладку **Мнемосхемы**. В списке мнемосхем нужно передвинуть нужную мнемосхему в начало списка с помощью кнопки ;
- кнопка **Звук.** Выключает звуковой сигнал, возникший при получении новой ошибки. Звуковой сигнал включится снова при получении другой новой ошибки;
- кнопка **Инфо.** Выдает окно с дополнительной информацией о выбранной ошибке. В окне отображается код ошибки, более подробное описание ошибки и инструкции по ее устранению;
- кнопка **Отчет.** Формирует отчет по всем ошибкам в списке. Пользователю предоставляется на выбор:

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

- 1 Печать отчета сразу на принтере;
 - 2 Предварительный просмотр отчета;
 - 3 Открытие отчета в виде документа Microsoft Office Word.
- флажок **Предупр.**. Указывает, выводить ли в списке предупредительные ошибки (аварийные выводятся всегда);
 - флажок **Квитир.**. Указывает, выводить ли в списке квитируемые ошибки (новые выводятся всегда);
 - флажок **Старые**. Указывает, выводить ли в списке старые ошибки. Флажок неактивен, если не указана настройка **Сигнализация – Хранить старые ошибки**;
 - флажок **НЛО**. Указывает, выводить ли в списке ошибки от неизвестного источника. Источник считается неизвестным, если его адрес не найден в БД. Это может случиться, в частности, если, в контроллер загружена не та технологическая программа;

3.3.4.5 Настройка

Для настройки окна ошибок выберите пункт **Настроить** его контекстного меню. В открывшемся диалоговом окне можно задать:

- **Цвет фона** списка ошибок;
- **Шрифт**, которым будут отображаться тексты ошибок;
- **Узел**. Выбор узла позволяет отфильтровать ошибки, идущие от смежного оборудования. Каждая ошибка относится к определенному узлу. Ошибка, связанная с маркой, относится к тому же узлу, что и эта марка. Ошибка, не связанная с маркой, относится к тому же узлу, что и абонент - источник ошибки. Если в окне настроек выбрать некоторый узел, то в списке будут отображаться только ошибки, относящиеся к этому узлу и его подузлам. Если выбрать **все узлы**, будут отображаться все ошибки, независимо от их принадлежности к узлам;
- **Опции отображения времени** задают, какие компоненты времени будут выводиться в столбцах **Появилась** и **Квитирувана**. Время состоит из 3-х компонент: дата, время суток и миллисекунды;
- **Фильтрацию ошибок** по их типу;
- **Видимость столбцов**;
- **Отображение разделителя суток**. Разделитель представляет собой отдельную строку в списке, в которой выводится дата. Строка разделяет ошибки, появившиеся в разные сутки. Разделитель суток уместно использовать, если время в списке выводится без даты;
- **Отображение панели инструментов** и кнопки ее скрытия;
- **Имя файла**, который служит шаблоном при печати списка ошибок. Файл должен быть шаблоном Microsoft Office Word.

3.3.4.6 Пример использования

В 4.5 показано, как с помощью окна ошибок сформировать панель **Приборной станции**.

3.3.5 Полис

3.3.5.1 Назначение

Данный элемент позволяет визуализировать ход выполнения логической программы в виде дерева этапов и шагов. Элемент использует ту же визуальную компоненту, что и Пилон при редактировании и обзоре алгоритма УЛП. Подробное описание работы с деревом Полиса приведено в руководстве по Пилону.

В отличие от Пилона, компонента не имеет панели управления и в ней заблокированы функции редактирования логической программы.

3.3.5.2 Настройка

Контекстное меню элемента «Полис» содержит следующие пункты:

- **Привязать марку** - вызывает диалоговое окно выбора марки (только для нетиповых изображений). Чтобы элемент работал, необходимо привязать к нему марку логической программы. В случае типового изображения (объектного окна УЛП) марка привязывается автоматически в оперативном режиме;
- **Базовый шрифт** - позволяет выбрать шрифт для отображения строк в дереве;

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						44


- **Фоновое выделение точек выполнения** - переключатель, определяющий, используется ли подсветка фоном при выделении текущего узла дерева.

3.3.6 Список параметров

3.3.6.1 Назначение







Элемент **«Список параметров»** отображает значения, и статус заданного набора технологических параметров в реальном времени в виде таблицы и возможно графика.

3.3.6.2 Особенности использования в Графите

Добавление и удаление параметров в список в режиме проектирования осуществляется по выбору пунктов контекстного меню **Добавить параметр(ы)** и **Удалить параметр(ы)** соответственно. При добавлении параметра открывается стандартное окно выбора параметра. При удалении, если число параметров больше одного, возникает окно удаления параметра, если параметр в списке один, то он удаляется. В окне удаления параметра задается порядковый номер параметра, начиная с 1 сверху вниз. Для удаления параметра нужно нажать на кнопку с символом , расположенную справа от поля номера параметра. Для удаления всех параметров нужно нажать соответственно на кнопку **Удалить все**. Для завершения удаления параметров нужно нажать на кнопку **ОК**.

3.3.6.3 Работа с элементом в оперативном режиме

В оперативном режиме статус параметра отображается в виде иконки, расположенной слева в графе параметра, и может принимать следующие значения:

- 1  Статус параметра ОК.
- 2  Нет значения параметра.
- 3  Параметр изменен.
- 4  Параметр изменен и послан.
- 5  Ошибка при получении значения параметра.
- 6  Ошибка при загрузке данных параметра из БД. Например, может появиться, если параметр был удален из БД или БД была повреждена. При появлении данной ошибки необходимо исправить БД или удалить поврежденные параметры из списка.

Если в настройках не была указана опция **Запретить редактирование в режиме выполнения**, то в оперативном режиме возможно добавление и удаление параметров при помощи соответствующих пунктов контекстного меню. Удаление параметров отличается от удаления в режиме проектирования. Для удаления необходимо выделить один или несколько параметров левым щелчком мыши. Для выделения нескольких параметров необходимо во время щелчка держать зажатой клавишу **Ctrl**. После выделения необходимо вызвать контекстное меню правым щелчком мыши на одном из выделенных параметров, после чего выбрать пункт **Удалить параметр** или **Удалить выделенное**.

Если в настройках была выбрана опция показа графика, то при помощи таблицы, можно задавать видимость кривых на графике, путем соответственно установки или снятия флажков, расположенных слева в графе параметра, левым щелчком мыши. Возможность добавления/удаления параметров при этом сохраняется.

Примечание - Все изменения списка параметров (добавление или удаление параметров), сделанные в оперативном режиме будут потеряны при закрытии мнемосхемы.

3.3.6.4 Настройка

Для настройки списка параметров выберите пункт **Настроить** его контекстного меню. В открывшемся диалоговом окне можно задать:

- цвет фона списка параметров;
- шрифт, которым будут отображаться значения параметров;
- выбрать интервал обновления значений параметров;
- выбрать видимость разделительных линий таблицы;
- установить запрет на добавление / удаление параметров в оперативном режиме;
- настроить состав столбцов;

Инва. № подл.	Подп. и дата	Взаим. инв. №	Инва. №	Подп. и дата

- задать расширенные опции отображения времени;
- задать видимость графика и его настройки (появляются после установки флага **Показывать график**).

3.3.6.5 Настройки графика

Если в настройках списка параметров активировать флаг **Показывать график**, то появится вкладка **Настройки графика**, где можно задать следующие параметры:

- 1 Тип графика, возможны варианты:
 - кривая;
 - гистограмма;
 - горизонтальная гистограмма;
 - точки;
 - круговая диаграмма;
 - площадь;
- 2 Объёмность графика с указанием атрибутов объёмного представления (перспектива, ортогональность и т.д.).
- 3 Цвет фона и поля графика, отображение градиентной заливки.
- 4 Заголовок над графиком с указанием атрибутов заголовка (выравнивание, цвет шрифта).
- 5 Состав параметров.
- 6 Отображение меток на графике, в которых обозначаются значения сигналов (или название параметров по выбору пользователя) в точках отсчета, с указанием цвета, прозрачности.
- 7 Подписи на осях, угол их поворота, позиция, признак инвертирования; заголовки осей, угол поворота.
- 8 В зависимости от выбранного типа графика на панели **Кривые** появляется соответствующая вкладка для задания настроек присущих выбранному типу графика (например, при выборе типа графика «Круговая диаграмма» появляется вкладка **Диаграмма**, где можно задать X-Радиус, Y-Радиус, вращение, отображение тени, «отколотовость» большей части).
- 9 Цвета кривых (если тип графика «Круговая диаграмма», то цвета секторов диаграммы и т.д.) их видимость.

На вкладке **Настройки графика** в самом низу есть флаг **Предварительный просмотр** управляющий видимостью графика предварительного просмотра.

Для активации настроек необходимо нажать кнопку **ОК** внизу окна, для отмены сделанных изменений кнопку **Отмена**, или кнопку **По умолчанию** для задания настройкам значений по умолчанию.

3.3.7 Состояния Архивных станций





3.3.7.1 Назначение

Элемент **«Состояния Архивных станций»** отображает состояние (доступность) всех Архивных станций проекта.

3.3.7.2 Работа с элементом в оперативном режиме

Элемент формирует список всех архивных станций проекта в виде двухуровневого дерева, где корнями являются основные архивные станции, а листьями соответствующие основной станции резервные.

Элемент циклически проверяет состояние архивных станций с частотой, заданной в настройках, и отображает его в виде иконки, расположенной слева от названия станции. Статус может принимать следующие значения:

- 1  Состояние станции в процессе определения.
- 2  Архивная станция доступна.
- 3  Архивная станция недоступна.
- 4  Архивная станция помечена как отсутствующая.

Дополнительные сведения о состоянии станции можно получить в виде подсказки, если выделить нужный узел древа станций левым щелчком мыши и задержать над ним курсор на 2 секунды.

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Инд. № подл.

Если в настройках была указана опция **Отображать кнопку «обновить»** то в нижней части элемента появится кнопка, при нажатии на которую состояние станций будет принудительно обновлено.

Если в настройках была указана опция **Оповещать об изменении статуса**, то, по изменении статуса любой из станций в списке, будет выведено окно оповещения, содержащее сведения об изменении статуса.

3.3.7.3 Настройка

Для настройки элемента выберите пункт **«Настроить»** его контекстного меню. В открывшемся диалоговом окне можно задать:

- цвет фона дерева станций;
- шрифт, которым будут отображаться имена станций;
- интервал обновления статуса станций;
- текст, отображаемый в дереве. Текст каждого элемента может содержать как имя станции, задаваемое в Администраторе БД, так и фактическое имя компьютера в Windows;
- видимость кнопки принудительного обновления состояния станций;
- необходимость вывода окна оповещения при изменении состояния станции.

3.3.8 Хроника дискретных событий

3.3.8.1 Назначение

Элемент **«Хроника дискретных событий»** отображает все существующие в технологической программе объекты, привязанные к библиотечному алгоритму ХДС (№ 87), с возможностью просмотра и печати дискретных событий для каждого из объектов в отдельности или по их совокупности. Данные дискретные события регистрируются контроллером в реальном масштабе времени согласно функциям алгоритма ХДС..

3.3.8.2 Настройка

Для настройки элемента **«Хроника дискретных событий»** выберите пункт **Настроить** его контекстного меню. В открывшемся диалоговом окне можно задать:




- режим вывода событий;
- период опроса контроллера;
- цвет фона событий;
- шрифт, которым будут отображаться тексты событий.

Примечание - Имеется возможность установить настройки элемента по умолчанию.




3.3.8.3 Работа с элементом в оперативном режиме

Элемент **«Хроника дискретных событий»** содержит список объектов ХДС, список событий, зафиксированных соответствующим одним или несколькими алгоритмами ХДС в зависимости от выбора режима вывода событий (отдельного или общего), и панель отображения момента первого события и инструментов вывода на печать списка событий.

В списке для каждого объекта отображается:

- состояние объекта с помощью символа. Возможны 3 состояния:
 - а)  — нет опроса контроллера;
 - б)  — идет опрос контроллера, объект ХДС – активен;
 - в)  — идет опрос контроллера, объект ХДС – неактивен;
- нумерация объектов в порядке следования с наименованием алгоритма и маркой объекта.

В списке для каждого события отображается:



- признак события с помощью символа. Возможны 3 признака:
 - а)  — первое событие;
 - б)  — последующие события;
 - в)  — идет опрос контроллера, объект ХДС – не активен;

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Изм Лист № докум Подп. Дата

- нумерация событий в порядке следования для конкретного алгоритма ХДС. Нумерация не зависит от способа вывода событий (отдельного или общего) и осуществляется только в пределах одного алгоритма ХДС;
- задержка (в мсек) от момента первого события, который также отображается на панели выше списка событий, или время регистрации события (в мсек) в зависимости от способа вывода событий - отдельного или общего;
- описание события. Текст, описывающий, **что** произошло;
- приемник события. Приемник события – объект алгоритма ХДС, который зафиксировал появление данного события.

Примечание - Приемник события отображается только при общем способе вывода событий для показа принадлежности события к тому или иному объекту алгоритма ХДС.

Панель инструментов содержит следующие элементы:


- Кнопка **Предварительный просмотр**  . Открывает отчет в виде документа Microsoft Office Word;
- Кнопка **Печать**  . Выдает отчет по всем событиям в списке сразу на принтер.

3.3.9 ActiveX

3.3.9.1 Назначение

ActiveX – это стандарт встраиваемых графических объектов в Microsoft Windows. Классы ActiveX могут создаваться различными производителями программного обеспечения. Каждый такой класс реализован в некотором программном модуле (обычно `dll` или `ocx`), который должен быть зарегистрирован в системном реестре. Некоторые классы регистрируются сразу при установке Windows, другие входят в состав различных пакетов, например, Microsoft Office или Adobe Acrobat. Несколько классов ActiveX входит в состав дистрибутива Квинта.

3.3.9.2 Рисование

Для добавления элемента ActiveX нажмите кнопку  и в выпавшем меню выберите пункт **ActiveX**. Будет выдано окно со списком всех зарегистрированных в системе классов ActiveX. Выберите нужный элемент из списка и установите элемент на требуемое место.

Примечание - Не все классы ActiveX могут быть использованы в Графите.

3.3.9.3 Настройка

Контекстное меню элемента ActiveX обычно имеет два специальных пункта:

- **Редактор свойств** – вызывает диалоговое окно, с помощью которого можно изменять свойства элемента. В выпадающем списке нужно выбрать свойство по имени и указать требуемое значение. Поддерживаются только логические, числовые и строковые свойства;
- **Библиотека типов** – генерирует список всех свойств и методов данного класса ActiveX, затем открывает список с помощью приложения «Блокнот» («Notepad»);

В контекстном меню могут также присутствовать пункты, характерные только для определенного класса ActiveX. В некоторых случаях элемент ActiveX предоставляет собственное контекстное меню, подавляющее меню Графита.

3.3.9.4 Пример

Отображение на мнемосхеме документа Adobe Acrobat. В списке зарегистрированных ActiveX выберите **Adobe Acrobat Control** и установите элемент на изображение. Вызовите редактор свойств и укажите в свойстве `src` имя файла с документом PDF. Документ будет выведен в поле элемента. Если требуется задать или изменить имя файла в оперативном режиме, устанавливайте это свойство во встроенной программе:

```
procedure OnOpen;
begin
    src := 'требуемое имя файла';
end;
```

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

4 Компонировка экрана

Даже если подготовить все необходимые изображения, при запуске операторской станции возникнет пустой серый экран. Для получения более осмысленного результата необходимо определить, как те или иные изображения будут появляться на экране операторской станции.

4.1 Рабочие столы

4.1.1 Понятие рабочего стола

Экран операторской станции включает в себя одну или несколько прямоугольных **панелей**, расположенных по краям экрана, и **рабочую область**.

Каждая панель состоит из одной мнемосхемы. Расположение панели задается в Графите и не меняется в процессе оперативной работы. Панель должна «прижиматься» к одной из четырех сторон экрана.

Рабочая область занимает оставшуюся часть в середине экрана. Внутри рабочей области попадают мнемосхемы, вызываемые оператором.

Рабочим столом называется определенная конфигурация панелей, задаваемая в Графите. При формировании рабочего стола для каждой панели указывается мнемосхема, которую содержит панель и расположение панели (сверху, снизу, слева или справа).

4.1.2 Редактор рабочих столов

Рабочие столы формируются в диалоговом окне (рисунок 13), вызываемом из Графита командой меню **Сервис / Редактор рабочих столов**.

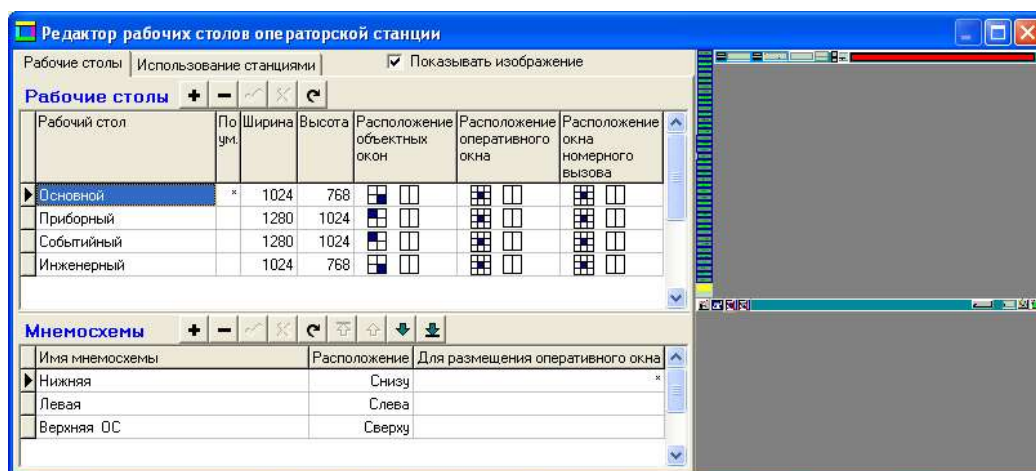


Рисунок 13 – Окно редактора рабочих столов. Закладка Рабочие столы

4.1.2.1 Закладка «Рабочие столы»

В этой закладке можно определить состав и свойства рабочих столов. Закладка содержит 2 таблицы:

- 1 Таблица **Рабочие столы** перечисляет сформированные рабочие столы. Кнопки + и – служат для добавления и удаления рабочих столов. Назначение столбцов таблицы:
 - а) **Рабочий стол** - здесь можно ввести имя, позволяющее идентифицировать рабочий стол;
 - б) **По ум.** – двойным щелчком левой кнопки мыши данный стол можно назначить **рабочим столом по умолчанию**;
 - в) **Ширина** – следует указать проектную ширину рабочего стола в пикселях;
 - г) **Высота** – следует указать проектную высоту рабочего стола в пикселях;
 - д) **Расположение объектных окон** – определяет начальное положение объектного окна при его открытии. Чтобы изменить этот параметр, кликните мышью в ячейку таблицы, а затем – в появившуюся кнопку «...»;

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

- е) **Расположение оперативного окна** – определяет позицию оперативного окна внутри панели, предназначенной для размещения оперативных окон;
 - ж) **Расположение окна номерного вызова** – определяет положение на экране окна, возникающего при вызове объекта по номеру.
- 2 Таблица **Мнемосхемы** перечисляет панели, входящие в состав рабочего стола, выбранного в верхней таблице. Кнопки + и – служат для добавления и удаления панелей, кнопки со стрелками – для изменения порядка панелей. Назначение столбцов таблицы:
- а) **Имя мнемосхемы** – отображает имя используемой в качестве панели мнемосхемы. Чтобы выбрать другую мнемосхему, кликните мышью в ячейку таблицы, а затем – в появившуюся кнопку «...»;
 - б) **Расположение** – указывает, к какой стороне экрана должна «прижиматься» данная панель;
 - в) **Для размещения оперативного окна** - двойным щелчком левой кнопки мыши можно указать, что оперативные окна должны помещаться внутрь этой панели.

4.1.2.2 Закладка «Использование станциями»

В этой закладке (рисунок 14) можно распределить рабочие столы по имеющимся мониторам. В таблице перечислены зарегистрированные в проекте компьютеры и указаны соответствующие им рабочие столы. Если к компьютеру подключено несколько мониторов и на них используются разные рабочие столы, следует выбрать этот компьютер в таблице и нажать кнопку **Добавить рабочий стол**. Для добавленной строки нужно указать номер монитора (например, 2) и соответствующий ему рабочий стол (4.2.6).

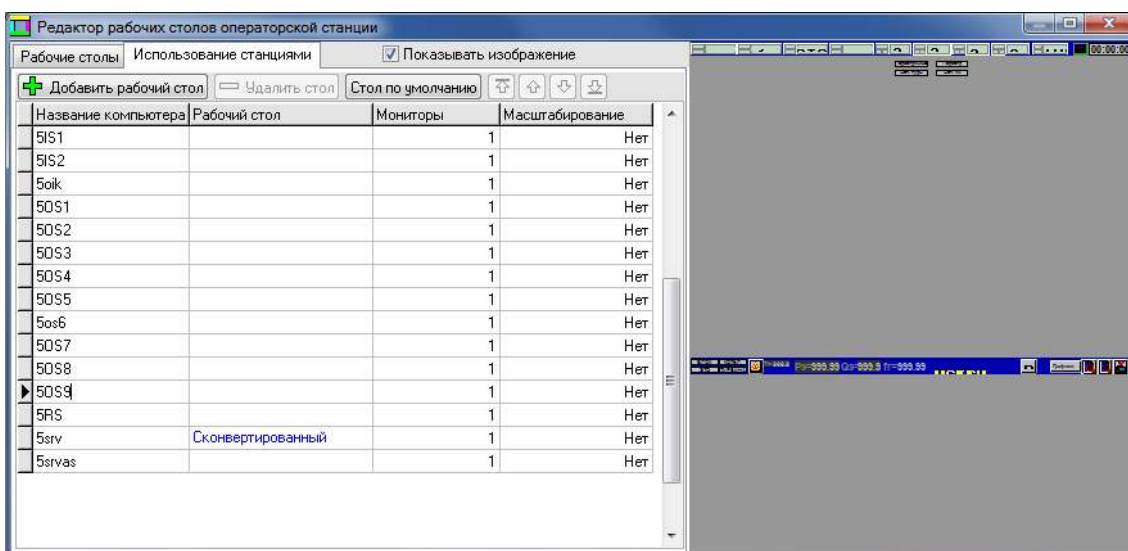


Рисунок 14 - Окно редактора рабочих столов. Закладка Использование станциями

Назначение столбцов таблицы:

- 1 **Название компьютера.** Это имя, под которым компьютер был зарегистрирован в Администраторе БД.
- 2 **Рабочий стол.** Это имя рабочего стола. Чтобы выбрать другой стол, кликните мышью в ячейку таблицы, а затем – в появившуюся кнопку «▼».
- 3 **Мониторы.** Это номера мониторов, на которые будет растягиваться выбранный рабочий стол. Для выбора мониторов, кликните мышью в ячейку таблицы, а затем – в появившуюся кнопку «...».
- 4 **Масштабирование.** Это способ растягивания рабочего стола на требуемый монитор (мониторы). Масштабирование применяется, если реальное разрешение монитора не совпадает с проектным разрешением рабочего стола (4.2.8).

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

Таблица 17 – Варианты масштабирования

Нет (по умолчанию)	Автоматическое масштабирование не производится. Параметры Шп и Вп игнорируются.
Непропорциональное	Обозначим Шм и Вм размеры монитора, на котором отображается данный рабочий стол. Тогда все мнемосхемы на этом мониторе будут растянуты в Шм / Шп раз по горизонтали и в Вм / Вп раз по вертикали. Если пропорции монитора не соответствуют проектным, т.е. Шм / Шп ≠ Вм / Вп, то мнемосхемы изменят свои пропорции, в частности, все окружности станут эллипсами.
Пропорциональное	Все мнемосхемы на мониторе размером Шм x Вм будут растянуты в К раз по горизонтали и по вертикали, где К = min(Шм / Шп, Вм / Вп). Если пропорции монитора не соответствуют проектным, то пропорции всех мнемосхем сохранятся, но слева или снизу возникнут незаполненные поля.

Если рабочий стол «растянут» на несколько мониторов, то в качестве Шм и Вм берутся размеры прямоугольной области виртуального экранного пространства, содержащего все эти мониторы. Например, если используются 2 монитора 1600x1200, расположенные рядом по горизонтали, то Шм = 3200 и Вм = 1200.

4.1.6 Переключение рабочих столов оператором

Как было сказано выше, при запуске операторской станции открывается рабочий стол, указанный в диалоговом окне **Редактор рабочих столов**. Проектант может предоставить оператору возможность сменить рабочий стол в оперативном режиме. Этой цели служит рецептор смены рабочего стола, который можно привязать к любому элементу изображения. При выполнении этого рецептора закрываются все мнемокадры, включая закрепленные, и открывается другой рабочий стол. Если к системному блоку подключено 2 монитора, и на каждом используется свой рабочий стол, то сменится только один из рабочих столов: тот, который содержит элемент изображения с этим рецептором.

4.2 Мнемокадры

Созданные в Графите изображения предоставляются оператору в виде мнемокадров. **Мнемокадр** – это окно, содержащее некоторое автономное изображение. Термин «Мнемокадр» введен в Квинте. Для пользователей, привычных к старой терминологии предыдущих версий Квинта, в таблице 17 приведены приблизительные соответствия терминов.

Таблица 18 – Соответствие терминов

Новый термин	Старый термин
Основной мнемокадр	Мнемосхема, окно переменного размера
Плавающий мнемокадр	Плавающее окно, рабочее окно
Закрепленный мнемокадр	-
Панель	Панель, системная мнемосхема
Оперативное окно	Оперативное окно

Операторская станция использует мнемокадры 3-х типов:

- основные;
- плавающие;
- закрепленные.

4.2.1 Основные мнемокадры

Основные мнемокадры всегда размещаются в **рабочей области** экрана и могут находиться в одном из трех состояний, в целом соответствующих состояниям окон Windows:

- развернутом;
- нормальном;
- свернутом.

Развернутый мнемокадр занимает всю рабочую область экрана, при этом он не имеет рамки и не может перемещаться. В развернутом состоянии может находиться только один мнемокадр.

В **нормальном состоянии** мнемокадр занимает часть рабочей области экрана, при этом мнемокадр имеет рамку с заголовком и может перемещаться в пределах рабочей области. Перемещая края или углы рамки, можно менять текущий размер листа. Заголовок рамки содержит

Инв. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инв. № подл.	

название изображения; справа и снизу расположены линейки, с помощью которых можно «прокручивать» изображение, если оно не умещается в рамке мнемокадра. Одновременно в нормальном состоянии могут находиться несколько мнемокадров.

Свернутый мнемокадр имеет минимальный размер: изображение скрыто, виден только заголовок рамки, мнемокадр можно перемещать. Одновременно в свернутом состоянии могут находиться несколько мнемокадров.

Имеется возможность оперативно менять состояние каждого мнемокадра – сворачивать его, переводить в нормальное состояние или разворачивать. Для этого используются специальные кнопки, расположенные на заголовке мнемокадра. Имеются специальные команды, которые позволяют нормальные листы расположить каскадом (с перекрытием) или в виде мозаики – вертикальной или горизонтальной.

На поведение основных мнемокадров влияет настройка **оконного режима (Квинтегратор – Настройка – Параметры – Операторская станция – Окна – Оконный режим)**. Имеются 3 оконных режима:

- однооконный;
- многооконный;
- смешанный (установлен по умолчанию).

В **однооконном** режиме может быть открыт только один основной мнемокадр, при этом он всегда находится в развернутом состоянии.

В **многооконном** режиме основные мнемокадры подчиняются стандартным законам многодокументной среды (MDI).



Смешанный режим отличается от многооконного тем, что:

- первый мнемокадр всегда открывается в развернутом состоянии;
- при разворачивании одного из мнемокадров все остальные мнемокадры закрываются.

4.2.2 Плавающие мнемокадры

Плавающие мнемокадры могут перемещаться оператором по всему экрану и имеют фиксированный размер. Они всегда расположены поверх основных и закрепленных мнемокадров. Плавающий мнемокадр имеет рамку с заголовком, содержащим название изображения.

Чаще всего в виде плавающих мнемокадров представляются объектные окна, т.е. типовые изображения некоторых объектов. Допускается выводить в плавающем виде также и мнемосхемы, обычно небольшого размера.

Плавающие мнемокадры могут автоматически выравниваться относительно друг друга и относительно границ рабочей области. Способ выравнивания можно настроить индивидуально для каждого рабочего стола. Для этого служит столбец **Расположение объектных окон** в диалоговом окне **Редактор рабочих столов**. К примеру, если задать выравнивание влево вверх с выстраиванием по горизонтали (что обозначается  ), то первый мнемокадр расположится в верхнем левом углу рабочей области, второй – справа от него и т.д. Оператор может перемещать мнемокадры как угодно, при этом можно восстановить исходное расположение с помощью рецептора «**Упорядочить объектные окна**».

Можно отключить автоматическое выравнивание плавающего мнемокадра при его открытии. Для этого вызовите из Квинтегратора приложение **Настройка – Параметры** и установите значение параметра **Операторская станция – Окна – Выравнивать при открытии** равном **Нет**. Эта настройка не влияет на работу рецептора «**Упорядочить объектные окна**».

4.2.3 Закрепленные мнемокадры

Закрепленные мнемокадры имеют фиксированный размер и располагаются в фиксированной области экрана. К ним относятся:

- панели, составляющие рабочий стол;
- оперативные окна.

Панели находятся на экране постоянно до тех пор, пока не будет закрыта операторская станция или не будет вызван рецептор смены рабочего стола. Их состав определяется проектантом и не может меняться оператором.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						53

Оперативное окно может открываться по команде оператора внутри одной из панелей. Для этой панели в редакторе рабочих столов должна стоять галочка в графе **Для размещения оперативного окна**. Оперативное окно выравнивается внутри этой панели в соответствие с настройкой «Расположение оперативного окна» для данного рабочего стола.

На рабочем столе может присутствовать только одно оперативное окно. Для открытия оперативного окна следует использовать рецептор открытия объектного окна. В настройке **Тип окна** для рецептора следует указать **Системное**, а в настройке события указать **Нажатие левой кнопки мыши**.

Оперативное окно автоматически закрывается при нажатии левой кнопкой мыши по любому месту экрана, к которому не привязан другой рецептор открытия оперативного окна.

4.3 Прочие окна

Кроме мнемокадров, в Операторской станции могут появляться окна, не содержащие изображений. Вид этих окон предопределен и не формируются проектным образом.

4.3.1 Окна сигнализации

Окно сигнализации содержит список всех имеющихся в Квинте в данный момент ошибок, удовлетворяющих определенным условиям, т.е. все технологические ошибки какого либо технологического узла (например узла «Котел»).

Окно сигнализации имеет рамку с заголовком, его можно перемещать по экрану и растягивать. Одновременно оператор может вызвать несколько окон сигнализации.

4.3.2 Диалоговые окна

Диалоговые окна характерны тем, что пока они открыты, остальные окна операторской станции недоступны для управления (тем не менее, их анимация продолжается). Диалоговое окно побуждает оператора выполнить некоторое действие, прежде чем продолжить работу с операторской станцией. Обычно, эти окна имеют кнопки **ОК** и **Отмена**, закрывающие окно.

Примером диалогового окна может служить окно ввода значения параметра, описанное в 5.3.

4.4 Сохранение состояния рабочего стола

Текущее состояние рабочего стола описывается набором открытых мнемокадров, основных и плавающих. Оператор меняет это состояние, открывая, закрывая, перемещая окна и изменяя их размер. Состояние рабочего стола можно сохранить на жестком диске. При запуске операторская станция восстановит состояние.

В приложении **Настройка - Параметры** нужно найти параметр **Операторская станция / Окна / Сохранять конфигурацию окон**. Допустимы следующие значения:

- **0** – состояние рабочего стола не сохраняется;
- **1** – сохраняется при выходе из операторской станции;
- **2** – сохраняется при каждом изменении.

На практике, отличие варианта 2 от 1 в том, что при нештатном выходе из ОС состояние, скорее всего, сохранится.

Если указано значение настройки 0 (т.е. состояние не сохраняется), можно задать начальное состояние, которое будет восстанавливаться каждый раз при запуске ОС. Для этого следует запустить Графит, открыть требуемые окна и вызвать пункт меню **Вид/Сохранить**. Конфигурация окон в Графите будет сохранена как начальная конфигурация для ОС.

Состояние рабочего стола сохраняется отдельно для каждого проекта, каждого компьютера и каждого рабочего стола на этом компьютере. При срабатывании рецептора смены рабочего стола происходит то же, что при перезапуске ОС: состояние старого рабочего стола сохраняется (если указано в настройках), затем восстанавливается сохраненное ранее состояние нового рабочего стола.

Конфигурация открытых окон в Графите также может автоматически сохраняться, это задается параметром **Графит / Окна / Сохранять конфигурацию окон**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

4.5 Реализация функций событийной и приборной станций

В Квинте 5 событийная и приборная станции были реализованы как отдельные приложения, запускаемые из Квинтегратора. Эти станции содержали фиксированный набор окон, отображавших информацию о событиях и ошибках.

В Квинте эти станции интегрированы в Операторскую станцию. Это позволяет:

- детально настраивать состав и расположение окон с событиями и ошибками;
- дополнять эти окна любой другой оперативной и архивной информацией;
- совмещать операторскую, событийную и приборную станции на одном рабочем месте.

В данном разделе мы опишем, как средствами Графита создать экраны, имитирующие внешний вид событийной станции Квинта 5.

4.5.1 Постановка задачи

Предположим, что на блочном щите под событийную станцию выделен компьютер с именем «Events». Требуется, чтобы при запуске операторской станции на компьютере «Events» экран имел бы компоновку, изображенную на рисунке 15.

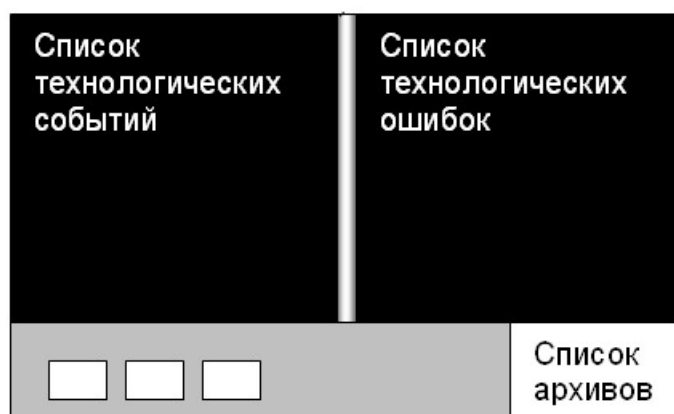


Рисунок 15 – Пример экрана для вывода событий

Список событий отображает последние технологические события в порядке их возникновения. Новые события автоматически добавляются в конец списка.

Список технологических ошибок содержит все актуальные на данный момент технологические ошибки.

Список архивов отображает имена и состояния доступности всех архивных станций проекта.

4.5.2 Подход к решению

- 1 Создается одна мнемосхема, которая будет занимать весь экран на компьютере «Events». Окна со списками добавляются на эту мнемосхему как элементы.
- 2 Создается отдельный рабочий стол, включающий только одну панель – эту мнемосхему.
- 3 Созданный рабочий стол назначается компьютеру «Events».

4.5.3 Создание мнемосхемы и рабочего стола

Пусть разрешение монитора на компьютере «Events» равно ШхВ. Для создания мнемосхемы рабочего стола требуется последовательно выполнить следующие действия:

- 1 Создадим новую мнемосхему, как описано в 2.3 и дадим ей имя «Событийная».
- 4 В диалоге **Файл / Параметры окна** укажем размер мнемосхемы равным ШхВ.
- 5 Добавим на мнемосхему элементы **Окно ошибок**, **Таблица данных из архива**, **Состояния Архивных станций** и другие вспомогательные элементы. Позиционируем элементы в точности так, как они должны располагаться на экране **Событийной станции**.
- 6 Откроем **Редактор рабочих столов**. В таблицу рабочих столов добавим новый стол, нажав кнопку «+». Для нового стола зададим имя **Событийный** и укажем ширину и высоту равными ШхВ.
- 7 Добавим мнемосхему **Событийная** к столу **Событийный**. Для этого нажмем кнопку «+» в таблице мнемосхем, затем нажмем кнопку «...» в столбце **Имя мнемосхемы** и выберем

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

мнемосхему **Событийная** в открывшемся диалоговом окне. В столбце **Расположение** следует выбрать любой из вариантов, например, **Сверху**.

- 8 В **Редакторе рабочих столов** перейдем на закладку **Использование станциями**. В таблице найдем компьютер «Events». (Если его там нет, это значит, что в Администраторе БД компьютер «Events» не прописан, или для него не указан флажок «Операторская станция».) В столбце Рабочий стол выберем для компьютера «Events» стол **Событийный**. Масштабирование задаем как **Непропорциональное**.

Теперь, при запуске операторской станции на компьютере «Events» будет открываться мнемосхема **Событийная**, занимающая весь экран.

4.5.4 Настройка списков

4.5.4.1 Список ошибок

В диалоге настроек списка (вызывается через контекстное меню **Настроить**) последовательно произвести следующие действия:

- 1 Выбрать цвет фона – черный, шрифт – такой, какой был выбран в **Администраторе БД** в опции **Стили и темы**.
- 9 Указать требуемый узел.
- 10 Поставить флажки **Отображать время, Отображать миллисекунды, Технологические, Текст ошибки, Источник ошибки, Отображать разделитель суток**. Остальные флажки убираем.

4.5.4.2 Список событий

Чтобы настроить список событий, следует воспользоваться контекстным меню **Настроить**. Далее произвести действия аналогичные при настройке списка ошибок (4.5.4.1).

4.5.4.3 Список архивов

Для настройки списка архивов следует также воспользоваться диалогом настроек списка (вызывается через контекстное меню **Настроить**) выбираем требуемый цвет фона и шрифт.

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

5 Задание динамики

В этой части документа описано, как настроить изображение так, чтобы отобразить на нем динамику процесса и реагировать на команды оператора.

5.1 Объекты и их параметры

5.1.1 Типы объектов

5.1.1.1 Простые

Слово **простой** применительно к типу объекта означает не **примитивно устроенный**, а не **содержащий других объектов**. Количество простых типов и набор их параметров предопределены в Квинте. Примерами простых типов являются Датчик аналоговый, Двигатель, Сигнал числовой и т.д.

5.1.1.2 Составные

Составные типы создаются проектантом в приложении **Администратор БД**. Составной тип всегда состоит из других типов (простых или составных). Объект составного типа не имеет собственных параметров: его параметрами являются параметры составляющих его объектов.

5.1.1.3 Специальные

Для расширения возможностей в Графите введены еще два специальных типа: **Узел** и **Система**. Объекты типа **Узел** соответствуют узлам проекта. Для узлов определены только сигнализационные параметры (см. ниже). Объект типа **Система** присутствует в одном экземпляре. Его параметрами являются свойства операторской станции в целом, например, режим работы, имя пользователя и т.д.

5.1.2 Типы параметров

В Квинте рассматривается 4 типа параметров. В таблице 19 приведены примеры.

Таблица 19 – Пример соответствия типов объектов и их параметров

Тип параметра	Описание	Пример	
		Тип объекта	Параметр
Число	Значение может меняться (непрерывно или дискретно) в определенном диапазоне. Частными случаями являются аналоговый и целочисленный параметры.	Задвижка универсальная	Положение
Состояние	Имеется допустимый набор значений, каждое значение поименовано. Частный случай – логический параметр с 2-мя значениями: Есть, Нет.	Задвижка универсальная	Состояние конечных
Время	Значение характеризует промежуток времени.	Задвижка универсальная	Тайм-аут
Набор бит	Значение представляет собой совокупность логических элементов.	Логическая программа	Условия

5.1.3 Свойства параметров

Текущее состояние объекта определяется значениями его параметров. Параметры характеризуются следующим набором свойств:


- **Значение** – текущее значение параметра. Точнее говоря, это последнее значение, которое было получено от источника этого параметра. Значение не определено, если параметр имеет статус **Нет значения** или **Ошибка**;
- **Предустановка** – значение, которое присваивается параметру пользователем (оператором) перед отправкой в контроллер. Например, пусть регулятор имеет задание 30%. Чтобы указать задание 50%, оператор устанавливает значение 50 для параметра **Задание**. При этом **Предустановка** параметра становится равной 50, но **значение** остается равным 30. Когда контроллер примет команду и операторская станция получит от него новое обновленное значение, тогда **значение** тоже станет равным 50. Обычно это происходит за доли секунд;

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

- **Статус** – состояние параметра. Принимает одно из значений:
 - а) **Норма** – значение параметра получено и совпадает с предустановкой;
 - б) **Нет значения** – значение пока не получено. Обычно этот статус имеют параметры при открытии мнемокадра (если они не использовались на других мнемокадрах). Через небольшое время статус **Нет значения** должен смениться на **Норма** или **Ошибка**;
 - в) **Изменен** – значение параметра изменено, но еще не послано в источник. Например, при управлении параметром с помощью ползунка, параметр будет иметь статус **Изменен**, пока оператор двигает ползунок, а при отпускании кнопки мыши его статус станет равным **Изменен и послан**;
 - г) **Изменен и послан** – значение параметра изменено и послано в источник;
 - д) **Ошибка** – значение параметра не может быть получено. Причиной может быть отсутствие указания источника в проекте, отсутствие связи с источником или ошибка, сформированная самим источником.
- **Управляемость** – указывает на возможность изменения параметра. Принимает значения «Нет» или «Да». Для управляемости необходимо, чтобы параметр был описан как управляемый в системной БД, чтобы у текущего пользователя были права на управление, и чтобы источник поддерживал функцию управления.
- **Качество** – дополнительная характеристика параметра, формируемая источником. Принимает одно из значений:
 - а) **Норма** – значение полностью корректно;
 - б) **Подмена** – реальное значение подменено пользовательским;
 - в) **Недостоверность** – значение определено, но недостоверно;
 - г) **Выход за уставку** – частный случай недостоверности, например, значение датчика «зашкалило»;
 - д) **Несравнение** – для резервированного значения количество «хороших» источников меньше требуемого;
 - е) **Прочее** – качество не может быть интерпретировано.

Качество не определено, если параметр имеет статус **Нет значения** или **Ошибка**.

5.1.3.1 Сигнализационные параметры

Для анимации в Графите можно использовать также сигнализационные параметры, обозначаемые в списке значком . Сигнализационный параметр говорит о наличии ошибки того или иного типа у данного объекта. Например, параметр **Техн.авар.новая** будет иметь значение «Есть», когда объект имеет хотя бы одну технологическую аварийную не квитированную ошибку.

В отличие от обычных параметров, сигнализационные параметры имеют следующие особенности:

- не запрашиваются у контроллера, а формируются на основании широковещательных сообщений об ошибках;
- не архивируются, поэтому имеют смысл только при работе с операторской станции с контроллерами;
- всегда имеют тип **Состояние** со значениями «Есть» и «Нет» и являются неуправляемыми.

Состав сигнализационных параметров фиксирован и не зависит от типа объекта.

Сигнализационные параметры определены также для узлов. Так, параметр **Техн.авар.новая** для узла будет иметь значение «Есть», если хотя бы один объект, входящий в этот узел, имеет технологическую аварийную не квитированную ошибку.

5.1.3.2 Выбор параметра объекта

Для целого ряда задач по определению состояния объектов требуется настроиться на одно из свойств того или иного параметра, например, при задании аниматоров и рецепторов. Для этой цели предусмотрены специальные диалоговые окна: для типовых изображений - **Выбор**







Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

параметра объекта и нетиповых изображений - **Выбор параметра и марки**, где помимо выбора параметра пользователю предоставляется следующие возможности:

- задать свойство параметра (по умолчанию - свойство **Значение**);
- вместо реальных значений параметра настроиться на нормативные, установив флажок **Норматив** (по умолчанию он сброшен);
- для параметра, представляющего собой набор бит, задать номер бита (по умолчанию – первый бит).

Поэтому, под выбором параметра объекта подразумевается выбор **определенного свойства реального или нормативного параметра**.

В списке параметров используются следующие мнемонические обозначения:

-  параметр типа **Число**;
-  параметр типа **Состояние**;
-  параметр типа **Время**;
-  параметр типа **Набор бит**;
-  параметр является управляемым;
-  параметр является сигнализационным.

5.1.4 Нормативные параметры

Кроме реальных значений параметров в Квинте можно использовать **нормативные значения** параметра. Общее понятие и методы работы с нормативными значениями описаны в документе «Комплексы программно-технические Квинт-6. Часть 1. Структура и системная интеграция. Руководство по эксплуатации ПФДИ.421457.003 РЭ1».

Предысторию и прогноз нормативного значения оператор может увидеть на графике.

Кроме того, текущее нормативное значение может быть выведено на изображении в любой форме, например в виде цифрового индикатора.

5.2 Анимация

Одной из основных задач Операторской станции является визуализация состояния технологического процесса. Чтобы «оживить» изображение, нужно связать его элементы с технологическими параметрами. Обычно для этого используют **аниматоры**. Аниматоры создаются и настраиваются в диалоговом режиме; они обеспечивают элементарную логику анимации. В случае более сложной логики анимации следует использовать встроенную программу, описанную в 5.5. Следует иметь в виду, что встроенная программа срабатывает после аниматоров.

5.2.1 Понятие аниматора

Чтобы элемент изображения изменял свой внешний вид при изменении каких-либо параметров, к этому элементу можно присоединить один или несколько аниматоров. Каждый аниматор связан с определенным параметром определенного объекта. В зависимости от значения параметра, аниматор изменяет определенный атрибут элемента, например, **Цвет заполнения** и **Цвет контура**.

Аниматоры можно классифицировать по следующим признакам:


- по тому, с какими типами технологических параметров можно их связать. Это - **аниматоры числовых параметров, аниматоры параметров состояний и наборов бит и аниматоры параметров времени**;
- по тому, на какие атрибуты элементов изображений они воздействуют. Это – **аниматоры раскраски, аниматоры текста, аниматоры общего назначения и специальные аниматоры**.

Особое место занимает группа **аниматоров значений параметров**. Аниматоры этой группы предназначены для отображения значений параметров в текстовом виде.

Список аниматоров можно увидеть в **Окне аниматоров**, открываемом кнопкой . В этом окне выводятся все аниматоры, присоединенные к выбранному элементу изображения. Если ни один

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

элемент не выделен, показываются аниматоры, присоединенные к **изображению в целом**. Окно аниматоров содержит ряд кнопок, с помощью которых можно создавать аниматоры.

Чтобы увидеть, какие элементы имеют аниматоры, откройте список элементов кнопкой . Элементы с аниматорами отмечены сиреневыми прямоугольниками. Цифра внутри прямоугольника означает количество аниматоров, присоединенных к данному элементу.

Итак, каждый аниматор связан с элементом изображения и с параметром объекта. Специфика работы аниматора зависит от типа параметра.

5.2.2 Редактирование аниматоров

Редактирование аниматоров производится в окне аниматоров, где на панели инструментов расположены кнопки, каждая из которых относится к определенной группе аниматоров. Аниматоры можно добавлять, удалять, копировать, вырезать, вставлять и перенастраивать, т.е. изменять параметр объекта и значения атрибутов элемента изображения. На аниматоры можно наложить систему приоритетов срабатывания.

Перед редактированием списка выберите элемент изображения и откройте окно аниматоров. Перечень доступных аниматоров зависит от атрибутов элемента изображения. Поэтому не все кнопки панели инструментов могут быть подсвечены.

5.2.2.1 Добавление аниматоров

Нажмите на кнопку требуемой группы аниматоров или по правой клавише мыши выберите пункт контекстного меню **Добавить**. Далее из меню выберите требуемый аниматор и в открывшемся диалоговом окне свяжите его с конкретным параметром. Привязка параметра к аниматору и задание свойства параметра описаны в 5.1.5.2.

Если значениям свойства выбранного параметра можно поставить в соответствие какое-либо значение аниматора, откроется диалоговое окно **Задание анимации**. Следует иметь в виду, что совсем необязательно каждому значению параметра ставить в соответствие значение аниматора. При помощи опции **Не использовать** можно пропустить некоторые значения. Тогда за этими значениями параметра сохранятся предварительно заданные атрибуты изображения.

5.2.2.2 Удаление аниматоров

Выберите аниматор и по правой клавише мыши выберите пункт контекстного меню **Удалить** или нажмите клавишу **Delete**. Чтобы удалить все аниматоры, по правой клавише мыши выберите пункт контекстного меню **Удалить все**.

5.2.2.3 Копирование аниматоров

Выберите аниматор и по правой клавише мыши выберите пункт контекстного меню **Копировать** или нажмите клавиши **Ctrl+Insert**. Вся информация об аниматоре попадет в буфер обмена.

5.2.2.4 Вырезание аниматоров

Выберите аниматор и по правой клавише мыши выберите пункт контекстного меню **Вырезать** или нажмите клавиши **Shift+Delete**. Аниматор будет удален, а вся информация о нем попадет в буфер обмена.

5.2.2.5 Вставка аниматоров

По правой клавише мыши выберите пункт контекстного меню **Вставить** или нажмите клавиши **Shift+Insert**. Аниматор попадет в список с данными из буфера.

5.2.2.6 Изменение параметра

Выберите аниматор и по правой клавише мыши выберите пункт контекстного меню **Параметр** или нажмите клавишу **F2**. В открывшемся диалоговом окне свяжите аниматор с другим параметром или другим свойством параметра. Привязка параметра к аниматору и задание свойства параметра описаны в 5.1.5.2.

5.2.2.7 Изменение значений атрибутов изображения

Выберите аниматор и по правой клавише мыши выберите пункт контекстного меню **Атрибуты** или нажмите клавишу **F3**. Откроется окно **Задание анимации**, где можно изменить настройку значений параметра на значения аниматора.

Инва. №	Подп. и дата
Инва. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инва. № подп.	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						60

5.2.2.8 Настройка приоритетов аниматоров

С одним и тем же атрибутом элемента изображения может быть связано несколько аниматоров. В таком случае действует следующая система приоритетов: если значение атрибута должно измениться под воздействием нескольких аниматоров одновременно, то значение атрибута изменит только тот аниматор, который в списке расположен первым. Если аниматоры воздействуют последовательно, то значение атрибута будет меняться согласно расположению аниматоров в списке, начиная с первого. Если в настройке аниматора, который расположен выше, для значения параметра указано **Не использовать**, то приоритет переходит к аниматору, стоящему в списке следующим.

Чтобы изменить приоритеты срабатывания аниматоров, воспользуйтесь кнопками панели приоритетов.

5.2.3 Аниматоры состояний и наборов бит

В таблице 20 перечислены аниматоры, которые можно связать с параметрами состояний и наборов бит.

Таблица 20 – Связь аниматоров с параметрами состояний и наборов бит

Аниматоры раскраски	Заполнение		Цвет заполнения	
			Мигание заполнения	
			Стиль заполнения	
	Контур		Цвет контура	
			Мигание контура	
			Стиль контура	
	Фон		Цвет фона	
			Мигание фона	
	Аниматоры текста	Текст		Цвет текста
			Мигание текста	
			Стиль текста	
Аниматоры значений параметров	Текст		Значение текста	
			Отетксовка	
			Цифровая индикация	
Аниматоры общего назначения	Видимость			
	Запрет			
Специальные аниматоры	Картинка		Номер картинки	
			Номер кадра	
			Период смены кадра	
	Перерисовка			
	Действие			

Доступность элементу изображения тех или иных аниматоров зависит от типа изображения и его атрибутов. Например, к элементу **Треугольник** нельзя присоединить аниматор **Текст / Цвет текста**, но можно присоединить аниматор **Заполнение / Цвет заполнения**, который в свою очередь можно связать с параметром **Состояние клапана Регулятора импульсного** и для каждого значения этого параметра (**Открыт**, **Закрыт** и др.) указать цвет заполнения треугольника. Сказанное справедливо и для таких свойств параметра объекта, как **Статус** и **Управляемость**, т.е. можно управлять значениями атрибутов элементов изображения, если значения указанных свойств связать с доступными аниматорами.

5.2.3.1 Аниматоры раскраски

Аниматоры раскраски воздействуют на атрибуты раскраски.

5.2.3.2 Аниматоры текста

Аниматоры текста воздействуют на атрибуты текста.

5.2.3.3 Аниматоры значений параметров

Значения параметров состояний и наборов бит можно отображать в виде меняющихся текстовых значений. Так, стандартные тексты значений параметров объектов, заложенные в системной базе данных, отображаются с помощью аниматора **Текст / Цифровая индикация**. Вместо стандартных текстов можно использовать тексты, введенные непосредственно в Графите - для этого предназначен аниматор **Текст / Значение текста**. Вместо стандартных текстов можно также использовать **отетксовки**, введенные в **Аркаде** – для этого предназначен аниматор

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Текст / Отетстовка. Аниматоры доступны элементу **Цифры** и только аниматор **Текст / Цифровая индикация** присоединяется к нему автоматически при размещении в выбранном месте.

Аниматор **Текст / Цифровая индикация** позволяет также отображать значения свойств **Статус** и **Управляемость** в виде стандартных текстовых значений, а аниматор **Текст / Значение текста** - в виде текстовых значений пользователя Графита.

5.2.3.4 Аниматоры общего назначения

Аниматоры общего назначения воздействуют на атрибуты общего назначения.

5.2.3.4.1 Аниматор видимости

В зависимости от значения параметра состояния или указанного бита аниматор позволяет сделать элемент изображения видимым или невидимым.

5.2.3.4.2 Аниматор запрета

Аниматор запрета позволяет временно заблокировать реакцию элемента на события мыши и клавиатуры.

В типовом случае этот аниматор присоединяют к элементу **Кнопка**, при этом можно задать, при каких состояниях выбранного объекта на кнопку можно нажимать, а при каких - нажатие кнопки будет заблокировано. В состоянии запрета команда управления формироваться не будет.

5.2.3.5 Специальные аниматоры

5.2.3.5.1 Аниматоры растровых изображений

Группа аниматоров **Картинка** связана с растровыми изображениями (картинками). С ее помощью в зависимости от значения параметра состояния или указанного бита можно менять одну картинку на другую, а для многокадровых картинок также изменять номер кадра или задать непрерывную последовательность кадров, определив частоту смены кадров (таким способом можно, в частности, получить эффект мультипликации).

5.2.3.5.2 Аниматор перерисовки

Аниматор можно связать с любым параметром объекта. Его следует применять, когда некоторый элемент "А" изображен поверх другого динамического элемента "Б". В этом случае требуется, чтобы при изменении элемент "Б" не затирал элемент "А". Для этого при каждом изменении элемента "Б" элемент "А" должен перерисовываться. Присоедините к элементу "А" аниматор перерисовки и свяжите его с тем параметром элемента "Б", изменение значения которого влияет на внешний вид элемента "Б".

5.2.3.5.3 Аниматор действия

Аниматор **Действие** позволяет автоматически выполнять те или иные действия, в зависимости от значения параметра состояния или указанного бита. Например, менять значения управляемых параметров, открывать и закрывать мнемосхемы и объектные окна, воспроизводить звуковые файлы и др. Этим достигается результат, аналогичный тому, к которому приводит использование рецепторов, т.е. решается задача оперативного управления технологическим процессом. Перечень допустимых действий, сгруппированных по категориям рецепторов, описан в 5.3.

5.2.4 Аниматоры числовых параметров

К числовым параметрам относятся аналоговые и целочисленные параметры. Ниже перечислены аниматоры, которые можно связать с числовыми параметрами, когда требуется только отображение значений свойств (таблица 21).

Таблица 21 – Связь аниматора с числовыми параметрами

Аниматоры значений параметров	Текст		Отетстовка	
			Цифровая индикация	
Специальные аниматоры	Шкала			
	Перерисовка			

Если требуется анимация изображений в зависимости от значений свойств **Статус** и **Управляемость**, воспользуйтесь аниматорами состояний и наборов бит. Например, при помощи аниматора **Контур / Цвет контура** можно управлять цветом контура элемента **Прямоугольник** в зависимости от статуса параметра **Значение сигнала Датчика аналогового**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

5.2.4.1 Аниматоры значений параметров

Значения свойств числовых параметров отображаются с помощью аниматора **Текст / Цифровая индикация**. В частности, свойство **Значение** обеспечит отображение текущего значения параметра, а свойство **Статус** – текущего значения статуса, например, “Норма” или “Ошибка”.

Для замены значений параметра на отекстовки пользователей Аркады предназначен аниматор **Текст / Отекстовка**.

Оба аниматора доступны элементу **Цифры** и только аниматор **Текст / Цифровая индикация** присоединяется к нему автоматически при размещении в выбранном месте.

5.2.4.2 Специальные аниматоры

5.2.4.2.1 Аниматор шкалы

Аниматор шкалы изменяет атрибут **Уровень шкалы**. Этот атрибут имеют такие элементы, как **Уровень**, **Ползунок** и **Движок**. Графический вид этих элементов плавно изменяется определенным образом при изменении данного атрибута.

Аниматор шкалы нормирует значение параметра так, чтобы оно лежало на отрезке [0, 1]. Границам этого отрезка соответствуют значения **Минимум** и **Максимум**, указанные для данного параметра в Аркаде.

5.2.4.2.2 Аниматор перерисовки

Назначение аниматора описано в 5.2.3..

5.2.5 Аниматоры параметров времени

Аниматоры числовых параметров, за исключением аниматора **Шкала**, можно связать с параметрами времени. В отличие от числовых параметров, параметры времени будут отображаться в виде интервала времени с указанием формата времени, например, 10 сек.


5.3 Управление


Помимо задачи визуализации состояния технологического процесса важнейшей задачей операторской станции является оперативное управление технологическим процессом. С одной стороны оперативное управление направлено на изменение значений **управляемых** параметров объектов, а с другой стороны – на выполнение предопределенных действий, например, открытие мнемосхем или объектных окон, управление расположением окон Операторской станции и др.

Оперативное управление реализуется посредством выполнения команд оператора. Для определения реакции (**рецепции**) операторской станции на команды оператора предназначены **рецепторы**, которые можно присоединить к любому элементу изображения или к изображению в целом. Рецепторы создаются и настраиваются в диалоговом режиме; они обеспечивают элементарную логику рецепции. В случае более сложной логики следует использовать встроенную программу, описанную в 5.4.

5.3.1 Понятие рецептора

Рецептор определяет действие, которое выполняется при наступлении какого-либо события. Например, можно задать перевод **Регулятора импульсного** на ручной режим работы при нажатии левой кнопкой мыши на элемент **Экранная кнопка**.

Чтобы определить, какие оперативные команды и в каком порядке выполнять, к элементу изображения можно присоединить один или несколько рецепторов. Список рецепторов можно увидеть в **Окне рецепторов**, открываемом кнопкой . В этом окне выводятся все рецепторы, присоединенные к выбранному элементу изображения. Если ни один элемент не выделен, показываются рецепторы, присоединенные к изображению в целом. Окно рецепторов содержит ряд кнопок, с помощью которых можно создавать рецепторы.

Чтобы увидеть, какие элементы имеют рецепторы, откройте список элементов кнопкой . Элементы с рецепторами отмечены синими прямоугольниками. Цифра внутри прямоугольника означает количество рецепторов, присоединенных к данному элементу.

Ив. № подл.	Подп. и дата
Взаим. инв. №	Ив. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						63

5.3.2 Редактирование рецептов

Редактирование рецептов производится в окне рецептов, где на панели инструментов расположены кнопки, каждая из которых относится к определенной категории рецептов. Ниже перечислены эти категории с указанием их назначения (таблица 22)

Таблица 22 – Категории рецептов

Параметр		Изменение значений управляемых параметров, квитирование ошибок и запоминание марок
Открыть		Управление открытием мнемосхем, объектных окон, списков ошибок и др.
Активное окно		Управление сверткой, разверткой и масштабом активных окон
Управление окнами		Управление расположением и закрытием окон
Звук		Управление включением и выключением звуков
Прочие		Вызов справки, настройка палитры, запуск процесса, настройка вида окна подтверждения и др.,

Рецепторы категории **Параметр** привязываются к управляемым параметрам объектов или к объектам в целом. Рецепторам других категорий такая привязка не требуется.

Рецепторы можно добавлять, удалять, копировать, вырезать, вставлять и перенастраивать. Действия, предопределенные рецепторами, выполняются в порядке расположения рецептов в списке. Перед редактированием списка рецептов выберите элемент изображения и откройте окно рецептов.

5.3.2.1 Добавление рецептов

Выберите требуемую категорию рецептов. Для этого нажмите соответствующую кнопку на панели инструментов или по правой клавише мыши выберите пункт контекстного меню **Добавить**. Далее из меню рецептов выберите требуемый рецепт. Вслед за этим, в большинстве случаев последует открытие диалоговых окон настройки рецептора. Работа с диалоговыми окнами описана в разделах документа, соответствующих каждому рецептору.

5.3.2.2 Привязка рецептора к событию

Чтобы при работе с операторской станцией оператор мог запускать соответствующие рецепторам команды, необходимо определить события, которые должны для этого произойти.

Для рецептов предусмотрены три группы событий: первая группа основывается на работе с левой клавишей мыши, вторая – на работе с клавиатурой и третья – на работе с контекстным меню по нажатию на элемент изображения правой клавишей мыши. Для каждого рецептора можно задать комбинацию событий по всем группам.

Выберите рецептор и вызовите пункт его контекстного меню **Событие** или нажмите клавишу **F4**. В открывшемся диалоговом окне **Задание события** (рисунок 16) укажите события, которые в оперативном режиме должны привести к выполнению команды, преопределенной выбранным рецептором.

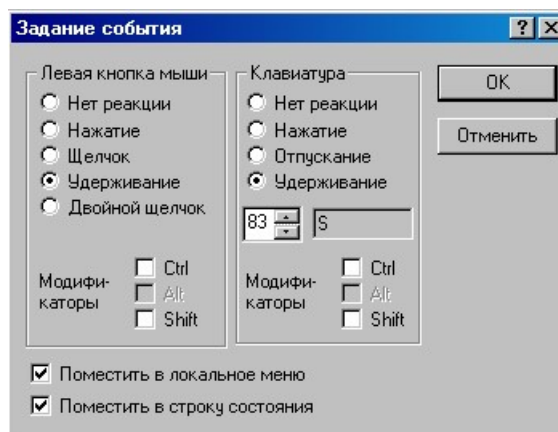


Рисунок 16 – Вид окна Задание события

Инд. № подл.	Инд. №	Взаим. инв. №	Подп. и дата	Подп. и дата

Для левой кнопки мыши предусмотрены следующие события:

- **Нажатие.** Команда выдается в момент нажатия левой кнопки мыши;
- **Щелчок.** Команда выдается в момент отпускания нажатой кнопки мыши (при нажатии и отпускании курсор должен находиться в зоне одного и того же элемента изображения);
- **Удерживание.** Команда периодически выдается все время, пока кнопка мыши удерживается нажатой и курсор находится в зоне одного и того же элемента изображения (обычно используется для изменения аналогового параметра);
- **Двойной щелчок.** Команда выдается с помощью двукратного нажатия кнопки мыши с коротким интервалом между нажатиями;
- **Нет реакции.** Команда выдаваться не будет.

При обработке события, связанного с мышью, вначале отработывает встроенная программа, затем – рецепторы.

Чтобы воспользоваться клавиатурой, нужно задать клавишу, нажатие которой вызовет выполнение требуемой команды. Клавиша задается либо ее числовым кодом, либо ее обозначением. Числовой код устанавливается в левом поле редактирования перебором допустимых значений от 0 до 255. При этом в правом поле будет отображаться соответствующее обозначение клавиши. Если поместить фокус в правое поле, щелкнув по нему мышью, можно задать обозначение клавиши, нажав на клавиатуре на требуемую клавишу. При этом в левом поле будет отображаться соответствующий числовой код клавиши.

Для клавиатуры предусмотрены следующие события:

- **Нажатие.** Команда выдается в момент нажатия на заданную клавишу;
- **Отпускание.** Команда выдается в момент отпускания нажатой клавиши (при нажатии и отпускании курсор должен находиться в зоне одного и того же элемента изображения);
- **Удерживание.** Команда периодически выдается все время, пока заданная клавиша удерживается нажатой и курсор находится в зоне одного и того же элемента изображения (обычно используется для изменения аналогового параметра). Период выдачи команд - не менее 100 мсек, но для некоторых рецепторов он может быть больше. Например, рецептор приращения аналогового параметра посылает команды в контроллер не чаще чем указано в Квинтеграторе в приложении Параметры\Операторская станция\Управление\Период посылки команд;
- **Нет реакции.** Команда выдаваться не будет.

Если в оперативном режиме открыто несколько изображений, то обработка нажатия клавиш будет выполняться в следующем порядке:

- 1 Системные объектные окна на каждом рабочем столе, начиная с главного.
- 2 Панели на всех рабочих столах.
- 3 Активное окно.

Нажатие левой клавиши мыши или кнопки клавиатуры можно сопроводить обязательным удержанием на клавиатуре клавиш **Ctrl** или **Shift** или обеих одновременно. Для этого нужно установить соответствующие флажки.

Привязывая рецептор к событию можно задать еще два свойства, которые проявляются в оперативном режиме работы, а именно: размещение рецепторов в локальном меню и в строке состояния.

Чтобы оператор мог выполнять команды управления из локального меню, нужно установить флажок **Поместить в локальное меню**. Тогда, если оператор выберет по правой кнопке мыши изображение, к которому присоединены рецепторы, на экране появится меню, где будут перечислены рецепторы, отмеченные этим флажком. После чего левой или правой кнопкой мыши можно выбрать нужную строку меню с последующим выполнением выбранной команды. Обычно этот флажок устанавливается, когда с одним изображением связано несколько рецепторов, которые должны выполнять команды по щелчку левой клавиши мыши. В оперативном режиме по наступлению этого события все эти команды будут последовательно выполняться. Поэтому, если по левой кнопке мыши нужно выполнить одну какую-то команду, то для рецепторов, выполняющих другие команды, следует указать **Нет реакции** и установить флажок **Поместить в локальное меню**.

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

Значения управляемых параметров позволяют изменять следующие рецепторы:

- Установка значения параметра;
- Установка отекстованного параметра;
- Набор значения параметра;
- Переключение дискретного параметра;
- Приращение аналогового параметра;
- Сброс предустановки;

Действия с объектами выполняют следующие рецепторы:

- Квитировать ошибки;
- Запомнить марку.

5.3.3.1 Рецептор установки значения параметра

Рецептор используется тогда, когда по команде оператора требуется установить заранее заданное в Графите значение **числового параметра, параметра состояний или параметра набора бит**. Например, для параметра **Режим работы Логической программы** задать значение **Информационный**. Вид диалоговых окон для установки значения параметра зависит от типа параметра. Если к рецептору привязан параметр состояний или набора бит, пользователь Графита выбирает значение параметра из списка стандартных значений в диалоговом окне **Выбор значения параметра** (рисунок 17).

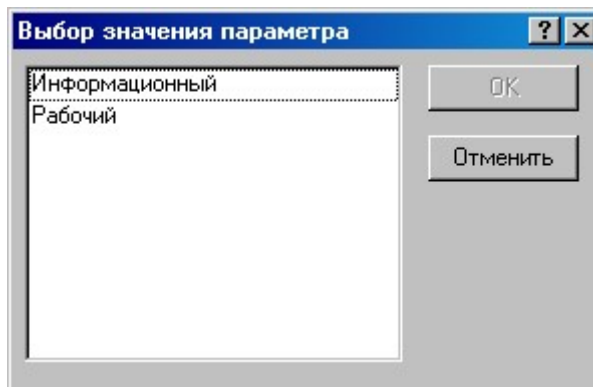


Рисунок 17 – Вид окна Выбор значения параметра

Если к рецептору привязан числовой параметр, пользователь Графита вводит значение параметра с помощью набора кнопок диалогового окна, название которого зависит от типа изображения и имени параметра. Если изображение типовое, названием окна является имя параметра, если нетиповое – марка объекта и имя параметра (рисунок 18).

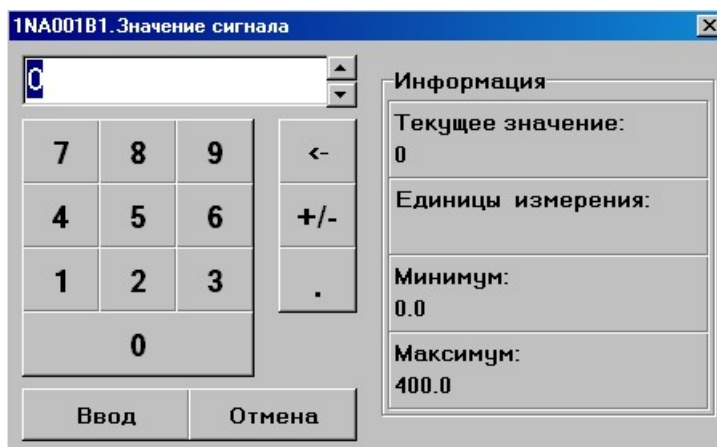


Рисунок 18 – Вид нетипового окна

Для ввода целой части значения аналогового параметра предназначены кнопки обозначенные цифрами от 0 до 9, а для десятых долей – пара кнопок верхнего окна редактирования. Для ввода

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

значений целочисленных параметров предназначены обе группы кнопок. Дополнительные кнопки предназначены для следующих целей:

- “<-” - для отмены введенной цифры;
- “+/-” - для изменения знака числа;
- “.” - для вставки разделителя целой и дробной части числа.

Значение параметра можно изменить. Для этого в окне рецепторов выберите требуемый рецептор и по правой клавише мыши - пункт контекстного меню **Значение**.

В окне рецепторов этот рецептор имеет вид: ‘Параметр->Значение’ или ‘Марка.Параметр->Значение’, например, ‘1NG10Log.Режим -> Рабочий или 1NG10Log.N этапа ->10’.

5.3.3.2 Рецептор установки отекстованного параметра

Рецептор используется тогда, когда по команде оператора требуется установить **отекстованное значение числового параметра, параметра состояний или параметра набора бит**. Если параметр отекстован пользователем Аркады, в оперативном режиме по предопределенному событию будет открываться диалоговое окно **Выбор значения**, в котором оператор из списка отекстованных значений параметра может выбрать требуемое значение.

В окне рецепторов этот рецептор имеет вид: ‘Параметр список’ или ‘Марка.Параметр список’, например, ‘1NG10Log.Режим список’ или ‘1NG10Log.N этапа список’.

5.3.3.3 Рецептор набора значения параметра

Рецептор используется, когда по команде оператора требуется ввести значение **числового параметра, параметра состояний или параметра набора бит**. Использует те же диалоговые окна, что и Рецептор установки значения параметра, но только в оперативном режиме, а не в Графите. Рецептор обладает свойством подавления выполнения следующих рецепторов. Активизация этого свойства производится в диалоговом окне **Свойства**, которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Свойства** (рисунок 19).

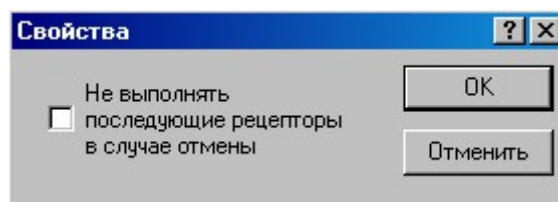


Рисунок 19 – Вид окна свойства

Если флажок **Не выполнять последующие рецепторы в случае отмены** установлен, то в оперативном режиме последующие рецепторы выполняться не будут, когда оператор нажмет кнопку **Отменить**.

В окне рецепторов этот рецептор имеет вид: ‘Параметр->?’ или ‘Марка.Параметр->?’.

Например, ‘1NG10Log.Режим ->?’ или ‘1NG10Log.N этапа ->?’.

5.3.3.4 Рецептор переключения дискретного параметра

Рецептор используется, когда по команде оператора требуется устанавливать альтернативные значения **параметра состояний или набора бит**. В этом случае оператор должен просто выполнять предопределенное действие. Например, нажимать левую клавишу мыши на элемент изображения **Двигателя двухскоростного**. Тогда, если к рецептору привязан параметр **Режим**, то при первом нажатии устанавливается автоматический режим работы, при следующем нажатии - ручной режим и далее по кругу.

В окне рецепторов этот рецептор имеет вид: ‘Параметр Значение1<->ЗначениеN’ или ‘Марка.Параметр Значение1<->ЗначениеN’, где Значение1 – первое значение параметра в диапазоне стандартных значений, а ЗначениеN – последнее.

Например, ‘1NA001B2.Режим Штатный<->Подмена’.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

5.3.3.5 Рецепт приращения аналогового параметра

Рецептор используется для плавного изменения **аналогового** параметра, например, параметра **Импульс Регулятора импульсного**. Для этого используется диалоговое окно **Изменение параметра**, которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Приращение** (рисунок 20).

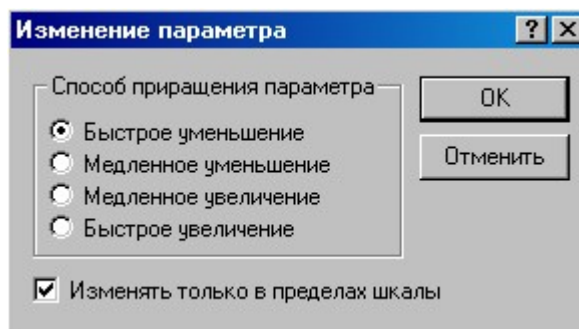


Рисунок 20 – Вид окна изменения параметра

Значения скоростей быстрого и медленного приращения в процентах диапазона задаются компонентой **Параметры** из папки **Настройка** основного окна **Квинтегратора**, где среди параметров настройки нужно выбрать узлы **Операторская станция / Управление / Скорость быстрого приращения** и **Операторская станция / Управление / Скорость медленного приращения**, а в этом окне можно выбрать один из четырех способов приращения параметра.

Если требуется, чтобы в процессе изменения параметр не выходил за диапазон шкалы, заданный в базе данных, можно установить флажок **Изменять только в пределах шкалы**.

В окне рецептов этот рецептор имеет вид: **'Параметр +'** или **'Марка.Параметр +'** для медленного увеличения, **'Параметр ++'** или **'Марка.Параметр ++'** для быстрого увеличения, **'Параметр -'** или **'Марка.Параметр -'** для медленного уменьшения и **'Параметр --'** или **'Марка.Параметр --'** для быстрого уменьшения, например, **'1NA001B2.Значение сигнала ++'**.

5.3.3.6 Рецепт сброса предустановки

Рецептор используется, когда по команде оператора требуется заменить значение предустановки реальным значением параметра. Предназначен для числовых параметров, параметров состояний и параметров набора бит.

В окне рецептов этот рецептор имеет вид: **'Параметр x'** или **'Марка.Параметр x'**, например, **'1NA001B2.Значение сигнала x'**.

5.3.3.7 Рецепт квитирования ошибок

Рецептор используется, когда по команде оператора требуется квитировать все ошибки из буфера ошибок контроллера. По predetermined действию квитироваться все ошибки, которые относятся к привязанному к рецептору объекту. Например, если к рецептору по конкретной марке привязан объект **Датчик аналоговый** и при работе с ним возникают какие-либо ошибки, то при нажатии на predetermined клавишу все ошибки будут квитированы.

В окне рецептов этот рецептор имеет вид: **'Квитировать ошибки'** или **'Квитировать ошибки (Марка)'**, например, **'Квитировать ошибки (1NA001B2)'**.

5.3.3.8 Рецепт запоминания марки

Рецептор используется, когда по команде оператора требуется запомнить марку объекта для дальнейшего использования её в трендах. По predetermined действию марка объекта копируется в буфер обмена. В дальнейшем сохраненную марку можно использовать, например, при добавлении кривой в элемент **«График»**.

В окне рецептов этот рецептор имеет вид: **'Запомнить марку'** или **'Запомнить марку (Марка)'**, например, **'Запомнить марку (1NA001B2)'**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

5.3.4 Рецепторы категории Открыть

В эту категорию входят рецепторы, с помощью которых в оперативном режиме можно вызывать на экран изображения, содержащие мнемосхемы, объектные окна, меню мнемосхем, списки ошибок и сборки. Для мнемосхем, меню мнемосхем и объектных окон требуется определить характеристики окон, в которые их изображения будут выводиться. В этих целях используется диалоговое окно **Характеристики окна** (рисунок 21), которое открывается автоматически при добавлении соответствующего нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Тип окна**.

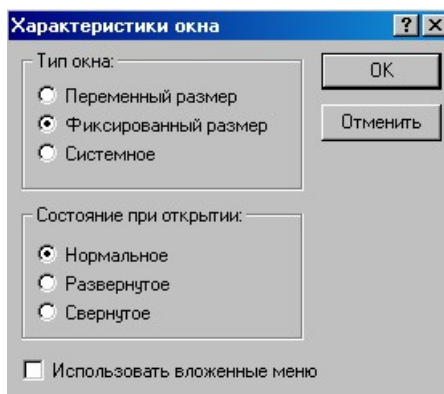


Рисунок 21 – Окно Характеристики окна

Флажок **Использовать вложенные меню** предназначен для окон с изображением меню мнемосхем. Если его включить, можно получить доступ к вложенным меню.

5.3.4.1 Рецепттор открытия мнемосхемы

Рецептор используется, когда по команде оператора требуется открыть заданную в Графите **мнемосхему**. Выбор мнемосхемы производится в диалоговом окне **Выбор мнемосхемы**, которое открывается автоматически при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Изображение**.

В окне рецепторов этот рецептор имеет вид: '**Наименование мнемосхемы**', например, 'Топливный тракт/мазут(1)'.
 Подп. и дата

5.3.4.2 Рецепттор открытия объектного окна

Рецептор используется, когда по команде оператора требуется открыть заданное в Графите **объектное окно**. Выбор объектного окна и привязка к нему конкретного объекта производится в диалоговом окне **Выбор изображения и марки**, которое открывается автоматически при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Изображение**.

В окне рецепторов этот рецептор имеет вид: '**Наименование объектного окна (Марка)**', например, 'Аналоговый датчик (1NA001B1)'.
 Инв. №

5.3.4.3 Рецепттор открытия объектного окна по выбору

Рецептор используется, когда по команде оператора требуется открыть **объектное окно**, тип которого задан в Графите, а марка будет выбрана в оперативном режиме. Выбор объектного окна производится в диалоговом окне **Выбор типового изображения**, которое открывается автоматически при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Изображение**.

Привязка к объектному окну конкретного объекта производится в оперативном режиме в диалоговом окне **Выбор марки**.

В окне рецепторов этот рецептор имеет вид: '**Окно "Наименование объектного окна" по выбору**', например, 'Окно "Аналоговый датчик" по выбору'.
 Взаим. инв. №

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

В окне рецептов он имеет вид: **‘Закрывать все мнемосхемы’**.

5.3.6.7 Рецепт закрытия всех объектных окон

Рецептор используется, когда по команде оператора требуется закрыть все объектные окна.

В окне рецептов он имеет вид: **‘Закрывать объектные окна’**.

5.3.6.8 Рецепт смены рабочего стола

Рецептор используется, когда по команде оператора требуется сменить Рабочий стол операторской станции. Привязка рецептора к рабочему столу производится в диалоговом окне **Выбор рабочего стола**, которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Рабочий стол**.

В окне рецептов этот рецептор имеет вид: **‘Рабочий стол “Имя стола”’**, например, ‘Рабочий стол “Стол №1”’.

5.3.7 Рецепторы категории Звук

В эту категорию входят рецепторы, с помощью которых в оперативном режиме можно управлять воспроизведением звуковых файлов. Чтобы обеспечить звуковое воспроизведение, операторская станция должна быть оборудована звуковой картой, соответствующими драйверами и звуковыми колонками. Кроме того, должны быть предварительно созданы звуковые файлы с расширением *wav*.

Простейшим редактором звуковых файлов может служить стандартная программа Sound Recorder (*sndrec32.exe*), которая поставляется вместе с Microsoft Windows XP. Можно воспользоваться и другими редакторами.

Все звуковые файлы должны лежать в определенной папке, которую мы будем называть **Папка звуков**. Путь к этой папке задается в приложении **Настройка / Параметры**, параметр **Звук / Папка звуков**.

Примечания

- 1 Если указать в качестве папки звуков сетевой ресурс на сервере БД, все операторские станции будут копировать звуковые файлы с сервера. Такой подход недопустим при промышленной эксплуатации, поскольку при отказе сервера звуковые файлы станут недоступны.
- 4 Вместо этого рекомендуется в качестве папки звуков указывать папку на локальном компьютере и средствами Windows копировать звуковые файлы на каждую операторскую станцию, использующую звуки.
- 5 Если данная Операторская станция не используется для выдачи звуков, следует для этой станции установить параметр настроек **Звук / Использовать** в значение «Нет». Тогда эта станция не будет искать и воспроизводить звуковые файлы.

5.3.7.1 Рецепт включения звука

Рецептор используется, когда по команде оператора требуется **воспроизвести звуковой файл**. Привязка рецептора к звуковому файлу производится в диалоговом окне **Включить звук** (рисунок 22), которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Звуковой файл**.

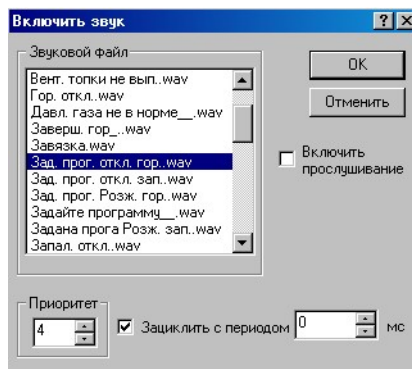


Рисунок 22 – Вид окна Включить звук

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

В этом окне можно выбрать конкретный звуковой файл и задать приоритет его воспроизведения. Звуки с более высоким приоритетом вытесняют звуки с более низким приоритетом. Дополнительно можно определить, должен ли выбранный звуковой файл воспроизводиться однократно или многократно с установленным интервалом до выключения. Для этого предназначен флажок **Зациклить с периодом** (рисунок 22) и окно, в котором можно задать период повтора в миллисекундах. В оперативном режиме период повтора может отличаться от установленного в Графите, поскольку звуковой файл должен быть воспроизведен, даже если период повтора меньше времени воспроизведения. Поэтому, период повтора принимает одно из двух значений: или продолжительность воспроизведения, или заданное в окне значение. Результат зависит от того, какое из этих значений больше. Если период повтора равен 0, звуковой файл будет воспроизводиться без пауз.

Звуковые файлы можно предварительно прослушивать в Графите, установив флажок **Включить прослушивание** (рисунок 22). Если при этом установлен флажок **Зациклить с периодом**, то воспроизведение будет повторяться многократно.

В окне рецепторов этот рецептор имеет вид: **‘Включить звук Имя звукового файла с приоритетом N’**, где N – заданный приоритет. Например, ‘Включить звук Запальник отключен.wav с приоритетом 6’.

5.3.7.2 Рецептор выключения звука

Рецептор используется, когда по команде оператора требуется **прекратить воспроизведение звукового файла**. Привязка рецептора к звуковому файлу производится в диалоговом окне **Выключить звук**, которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Звуковой файл**.

В окне рецепторов этот рецептор имеет вид: **‘Выключить звук Имя звукового файла’**. Например, ‘Включить звук Запальник отключен.wav с приоритетом 6’.

5.3.7.3 Рецептор выключения звуков по приоритету

Рецептор используется, когда по команде оператора требуется **прекратить воспроизведение звуковых файлов с одинаковым приоритетом**. Настройка рецептора на значение приоритета воспроизведения звуковых файлов производится в диалоговом окне **Выключить звук** (рисунок 23), которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Звуковой файл**.

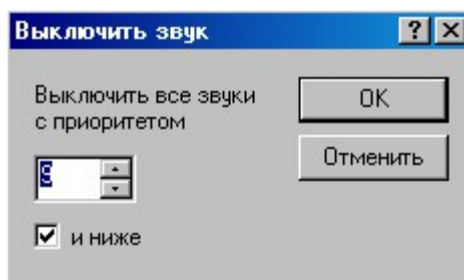


Рисунок 23 - Окно Выключить звук

В окне **Выключить звук** можно задать приоритет воспроизведения звуковых файлов и установить флажок **и ниже**. Тогда по predetermined действию оператор может прекратить воспроизведение тех звуковых файлов, приоритет которых меньше или равен заданному. Если флажок **и ниже** не установлен, то прекращается воспроизведение только тех файлов, приоритет которых равен заданному.

В окне рецепторов этот рецептор имеет вид: **‘Включить звуки с приоритетом N и ниже’** или **‘Включить звуки с приоритетом N’**, где N – заданный приоритет. Например, ‘Включить звуки с приоритетом 7 и ниже’.

5.3.7.4 Предопределенные звуки сигнализации

Операторская станция может выводить звуки не только в результате срабатывания заданного в Графите рецептора, но и «по своей инициативе», при получении широкоэвещательных ошибок. При

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

получении новой ошибки автоматически запускается один из указанных в таблице 23 файлов, в зависимости от типа ошибки. (Указанные файлы также должны находиться в папке звуков.)

Таблица 23 – Соответствие звуков типам ошибок

Тип ошибки	Имя звукового файла	Приоритет
Приборная аварийная	alarm11.wav	7
Приборная предупредительная	alarm12.wav	3
Технологическая аварийная	alarm21.wav	7
Технологическая предупредительная	alarm22.wav	3
Сетевая аварийная	alarm41.wav	7
Сетевая предупредительная	alarm42.wav	3
Защитная аварийная	alarm81.wav	7
Защитная предупредительная	alarm82.wav	3

Для выдачи звуков сигнализации используется та же очередь, что и для звуков, выдаваемых рецепторами. Таким образом, если рецептор выдал зацикленный звук с приоритетом, например, 5, то этот звук будет «перебит» звуком аварийной сигнализации (приоритет 7), но не будет перебит звуком предупредительной (приоритет 3).

При исчезновении всех неkwитированных ошибок определенного типа соответствующий звук удаляется из очереди. При нажатии кнопок **Квит** или **Звук** в окне сигнализации из очереди удаляются все звуки alarmNN.wav, запущенные сигнализацией, но остаются звуки, запущенные рецепторами.

5.3.8 Рецепторы категории Прочие

В эту категорию входят рецепторы, которые по своему назначению не вписываются ни в одну из вышеперечисленных категорий.

5.3.8.1 Рецептор настройки палитры

Рецептор используется, когда по команде оператора требуется вызвать диалоговое окно **Настройка палитры**, работа с которым описана в 6.1.1.

В окне рецепторов он имеет вид: **‘Изменение палитры’**.

5.3.8.2 Рецептор полной перерисовки

Рецептор используется, когда по команде оператора требуется выполнить полную перерисовку экрана, т.е. вернуть изображение к первоначальному виду.

В окне рецепторов он имеет вид: **‘Перерисовка’**.

5.3.8.3 Рецептор подтверждения

Рецептор используется, когда оператору нужно предоставить возможность продолжать или подавлять выполнение рецепторов, следующих за этим рецептором. Для этого, по предопределенному действию, открывается окно подтверждения, вид которого настраивается в диалоговом окне **Вызов окна подтверждения** (рисунок 24). Оно автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Свойства**.

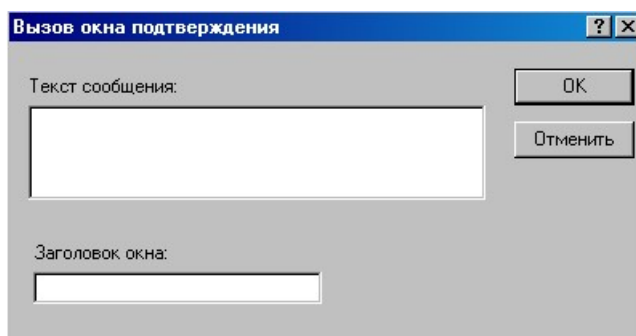


Рисунок 24 – Вызов окна подтверждения

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

Например, если по команде оператора требуется отключить воспроизведение звуковых файлов, то перед рецептором **Выключить звуки по приоритету** можно расположить рецептор **Подтверждение**, настроив его следующим образом: в окне **Текст сообщения** ввести текст “**Вы действительно хотите отключить воспроизведение звуковых файлов?**”, а в заголовок окна - “**Подтвердите ваши действия**” (рисунок 25). Тогда в оперативном режиме по команде оператора откроется соответствующее окно.

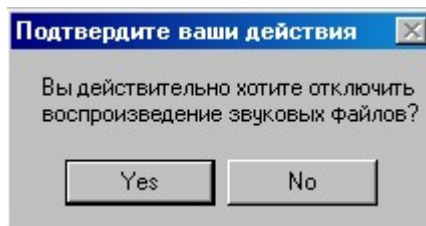


Рисунок 25 – Окно Подтвердите ваши действия

Если оператор нажмет на кнопку **Yes**, рецептор **Выключить звуки по приоритету** и все последующие выполняются, а если на кнопку **No** – рецепторы проигнорируются.

В окне рецептов этот рецептор имеет вид: ‘**Подтверждение**’.

5.3.8.4 Рецептор выхода

Рецептор используется, когда по команде оператора **требуется завершить работу операторской станции**. Если включена система защиты доступа к операторской станции, то по предопределенному действию открывается диалоговое окно **Выход**, в котором оператор может указать одно из предусмотренных действий по завершению работы. Окно имеет вид, показанный на рисунке 26.

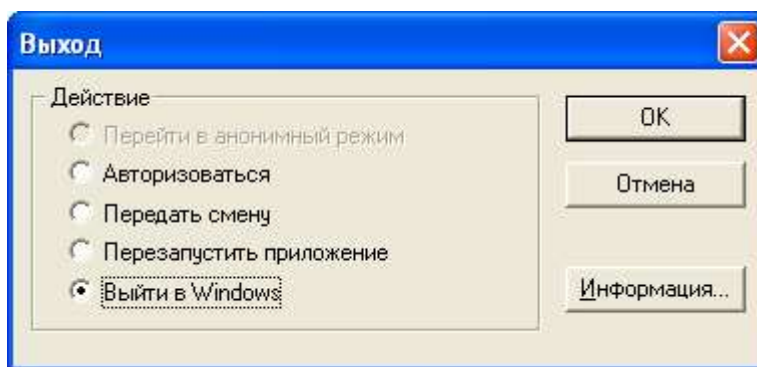


Рисунок 26 – Окно Выход

Если система защиты выключена, диалоговое окно **Выход** (рисунок 26) не открывается, но, тем не менее, работа операторской станции завершается и управление передается операционной системе Windows.

В окне рецептов этот рецептор имеет вид: ‘**Общий выход**’.

5.3.8.5 Рецептор вызова справки

Рецептор используется, когда по команде оператора **требуется вызвать файл-справку**. Рецептор предназначен только для файлов с расширением .hlp, которые поддерживаются программой Windows Winhlp32 Stub (Winhlp32.exe). Настройка рецептора на требуемый файл производится в диалоговом окне **Вызов справки** (рисунок 27), которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Свойства**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

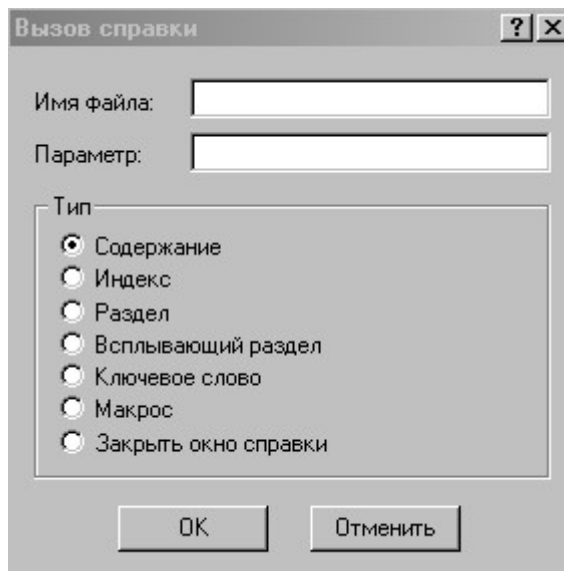


Рисунок 27 – Окно Вызов справки

В этом окне задается имя файла-справки с полным путем доступа к нему, тип требуемой информации и параметры поиска информации. Расширение .hlp можно не задавать. Параметры требуются не для всех типов информации. Параметр требуется для поиска разделов, всплывающих разделов, ключевых слов и макросов. Например, если требуется найти раздел, параметром будет служить имя раздела.

В окне рецепторов этот рецептор имеет вид: 'Справка'.

5.3.8.6 Рецепт запуска процесса

Рецептор используется, когда по команде оператора требуется **запустить процесс** (приложение системы Windows). Настройка рецептора на требуемый процесс производится в диалоговом окне **Запуск процесса** (рисунок 28), которое автоматически открывается при добавлении нового рецептора или при выборе по правой клавише мыши пункта контекстного меню **Свойства**. Окно имеет следующий вид:

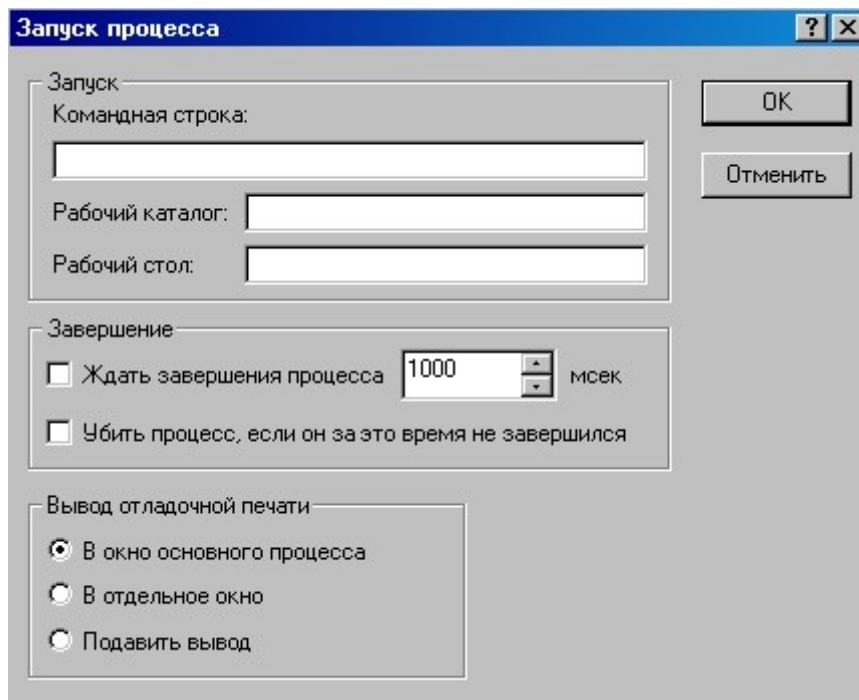


Рисунок 28 – Окно Запуск процесса

В поле **Командная строка** задается имя EXE-файла запускаемого приложения и параметры командной строки (если они предусмотрены). Если EXE-файл находится в папке **Bin** Квинта или в

Имп. №	Подп. и дата
Взаим. имп. №	Подп. и дата
Имп. № подл.	Имп. №

папке, входящей в переменную среды PATH, достаточно указать имя файла без пути. В противном случае необходимо указать полный путь. Начиная с версии 6.6, в командной строке можно использовать переменные среды в виде %varname%, например:

```
%ProgramFiles%\Microsoft Office\Office11\winword.exe
```

Поле **Рабочий каталог**, как правило, можно не заполнять. При этом рабочий каталог будет совпадать с тем каталогом, в котором находится EXE-файл.

В поле **Рабочий стол** можно задать либо имя рабочего стола операционной системы Windows, либо **Имя рабочей станции / Имя рабочего стола Windows**. Если это окно оставить незаполненным, приложение запустится на той же станции и на том же рабочем столе Windows, что и Операторская станция.

Если установить флажок **Ждать завершения процесса**, операторская станция приостановит работу до тех пор, пока процесс не завершится. Время ожидания можно ограничить, задав таймаут в миллисекундах.

Примечание - Не рекомендуется устанавливать флажок **Ждать завершения процесса** без особой необходимости, т.к. это приводит к временной неработоспособности операторской станции.

Если запускаемое приложение использует стандартный текстовый вывод, этот вывод можно подавить или направить в одно из двух предусмотренных окон с помощью группы радио кнопок **Вывод отладочной печати**.

В окне рецепторов этот рецептор имеет вид: **'Запуск приложения'**.

Например: Выключение компьютера. В Графите нет рецептора выключения компьютера. Если в операторской станции требуется такая кнопка, можно использовать рецептор запуска процесса. В папке Windows\system32 имеется приложение shutdown.exe, которое позволяет, в частности, произвести выключение компьютера. В поле **Командная строка** настроек рецептора укажите:

```
shutdown.exe -s -f -t 0
```

Остальные поля не заполняйте. Параметры командной строки можно уточнить, набрав **shutdown /?** в окне cmd.

Рекомендуется также вставить рецептор подтверждения перед рецептором, запускающим shutdown.exe.

5.4 Встроенная программа

Встроенная программа - это текст на специальном языке (называемый исходным текстом программы), который может быть привязан к любому элементу изображения или к изображению в целом. Если текст правильно составлен, то он может быть откомпилирован в специальный псевдокод, который затем будет выполняться при запуске изображения в реальном режиме или с эмулятором. Исходный текст и полученный псевдокод сохраняются на диске вместе с изображением.

5.4.1 Для чего нужна программа

5.4.1.1 Альтернативный способ анимации изображений

Вместо задания большого количества аниматоров для элемента изображения достаточно написать одну процедуру OnDataChange в исходном тексте программы, связанной с этим элементом. На языке программы можно выразить любой аниматор и часть рецепторов.

5.4.1.2 Расширение возможностей анимации

С помощью программы можно задать значительно более сложную связь между параметрами объектов и атрибутами элементов изображений. Например, можно добиться, чтобы цвет элемента менялся в зависимости от того, возрастает данный аналоговый параметр или убывает.

5.4.1.3 Возможность автоматического управления

Из программы можно изменить значение любого управляемого параметра объекта либо при какой-либо команде оператора, либо автоматически при изменении других параметров, либо по таймеру, либо по комбинации вышеуказанных событий.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						78

5.4.1.4 Возможность подключения внешних библиотек

Из программы можно вызывать зарегистрированные функции из динамически подключаемых библиотек (DLL). Зарегистрированы стандартные математические функции. Можно также зарегистрировать функцию из DLL, написанной пользователем.

5.4.1.5 Взаимодействие с элементами ActiveX

Динамически изменять свойства элементов ActiveX с помощью аниматоров нельзя. Для этого следует использовать встроенную программу.

5.4.2 Синтаксис

Формальное описание синтаксиса в виде диаграмм приведено в 7.2.12. В целом язык представляет собой упрощенный вариант языка Pascal с возможностью использования русских букв и расширений типа имени параметра объекта или атрибута примитива.

Лексемами (словами) языка являются:

- ключевые слова (const, var, procedure, function, begin, end, if, then, else, case, of, double, result);
- идентификаторы (слова из цифр или букв русского или английского алфавита, первый символ - обязательно буква, допускается также символ подчеркивания);
- числовые константы;
- произвольные строки в одинарных кавычках;
- операторы и разделители.

Маленькие и большие буквы в программе не различаются.

Комментарии заключаются в фигурные скобки {}.

5.4.3 Как устроена программа

Каждая программа привязана к определенному элементу изображения или к изображению в целом. Программы, относящиеся к разным элементам, не могут напрямую взаимодействовать друг с другом. Программа состоит из набора описаний констант, переменных, процедур и функций.

5.4.3.1 Константы

Константы являются просто символическим обозначением чисел. Если в формуле встречается некоторая физическая величина (время, температура и т.д.), рекомендуется вместо прямого указания величины ввести именованную константу – формула станет понятнее. Это особенно важно, если одна и та же величина входит в формулы несколько раз. Если некоторое выражение состоит из одних констант, его значение будет вычислено еще на этапе компиляции в Графите, а операторская станция будет работать с ним как с обычным числом. Например:

```
{ Расчет силы давления на заслонку в цилиндрической трубе }
const
    Diam = 0.3; { диаметр трубы, м }

function СилаДавления(Давление: double): double;
begin
    Result := Давление * Pi * Diam * Diam / 4;
end;
```

5.4.3.2 Глобальные переменные

Глобальные переменные создаются и обнуляются при открытии мнемокадра, содержащего элемент с программой, и уничтожаются при закрытии мнемокадра. Значения глобальных переменных могут быть изменены в любой процедуре или функции данной программы. Глобальные переменные из программ, относящихся к разным элементам изображения, никак не связаны между собой. Глобальные переменные описываются вне функций и процедур. Например:

```
var
    ПрошлоВремени: double; { время (сек), прошедшее от пуска двигателя }
```

Инд. № подл.	
Подп. и дата	
Взаим. инв. №	
Инд. №	
Подп. и дата	
Взаим. инв. №	

5.4.3.3 Локальные переменные

Локальные переменные описываются внутри функции или процедуры. Они создаются при входе в функцию и уничтожаются при выходе. Локальные переменные можно использовать только в той функции (или процедуре), в которой они были описаны. Обычно их используют для хранения промежуточных результатов расчета.

5.4.3.4 Функции и процедуры

Представляют собой фрагменты кода программы. Функция отличается от процедуры только тем, что она возвращает некоторый результат, а процедура нет. Из исходных функций и процедур можно обращаться к другим процедурам, функциям, переменным или константам, если последние находятся в той же программе и описаны выше в тексте этой программы.

5.4.3.5 Обработчики событий

Обработчик события представляет собой процедуру с одним из заранее определенных имен: **OnOpen**, **OnClose**, **OnDataChange**, **OnMouse**, **OnKey**, **OnTimer**. Обработчик события является «точкой входа» в программу, т.е. его вызывает среда операторской станции, а не другая процедура или функция. К примеру, обработчик **OnOpen** вызывается при открытии мнемокадра, содержащего элемент с программой.

5.4.3.6 Типы данных

Встроенная программа поддерживает 2 типа данных: число (double) и строка (string).

Тип **double** представляет числа с плавающей точкой в диапазоне от -10^{308} до 10^{308} .

Тип **string** представляет строки произвольной длины.

5.4.4 Что может делать программа

5.4.4.1 Доступ к параметрам объектов

Чтобы получить текущее значение объектного параметра, используйте выражение типа

```
Объект ('NA004B1').ЗначениеСигнала
```

В этом примере 1NA004B1 – это марка объекта «Аналоговый датчик», а «Значение Сигнала» – его параметр.

Значение аналогового параметра представляется в физических единицах. Для параметра типа «Состояние» каждое значение является поименованной константой, которые представляются следующим образом:

```
Объект ('NA004B1').Режим.Подмена
```

Чтобы изменить значение параметра, используйте оператор присваивания, например:

```
Объект ('NA004B1').Режим := Объект ('NA004B1').Режим.Подмена;  
Объект ('NA004B1').ЗначениеСигнала := 400;
```

Если программа обносится к элементу типового изображения, префикс Объект('...') можно опустить; тогда параметр автоматически свяжется с нужной маркой. Приведенные выше примеры для объектного окна аналогового датчика будут выглядеть так:

```
Режим := Режим.Подмена;  
ЗначениеСигнала := 400;
```

Для объекта составного типа в выражение следует дополнительно вставить имя члена составного типа. Предположим, составной тип «Холодильник» имеет член «Температура Камеры» типа «Датчик аналоговый». Тогда в программе для объектного окна холодильника можно получить температуру в камере как

```
ТемператураКамеры.ЗначениеСигнала
```

Если, в свою очередь, объект «Склад» содержит член «Холодильник15», то его температура доступна как

```
Холодильник15.ТемператураКамеры.ЗначениеСигнала
```

Значение параметра может быть недоступно на момент выполнения программы (например, если нет связи с контроллером). В этом случае, при попытке вычислить значение параметра

Подп. и дата				
Инв. №				
Взаим. инв. №				
Подп. и дата				
Инв. № подл.				
ПФДИ.421457.009 И3.5				
Лист				
80				
Изм	Лист	№ докум	Подп.	Дата

выполнение программы прерывается (до следующей точки входа), и на консоль выводится сообщение об ошибке. Если такое поведение неудовлетворительно, проверяйте статус параметра перед обращением к его значению:

```
if ЗначениеСигнала.Статус = Норма then
    Write (ЗначениеСигнала);
```

Подробнее о доступе к параметрам см. в 7.2.

5.4.4.2 Доступ к атрибутам элемента изображения

Программа может получать и изменять атрибуты элемента, к которому она относится. Примеры:

```
ЦветЗаполнения := Синий;
ЛеваяГраница := ЛеваяГраница + 10;
```

Перечень атрибутов приведен в разделе 7.

5.4.4.3 Доступ к данным проекта

Из программы можно обращаться не только к текущему состоянию объекта, но и к его статическим свойствам, описанным в проекте. Для аналогичных целей служит элемент «Реквизит», но программа предоставляет в этом плане более широкие возможности.

Поскольку проект хранится в таблицах базы данных (или в упакованном dat-файле), то доступ к проектным данным производится по именам таблиц и полей. В таблице 24 можно видеть несколько примеров для объектного окна аналогового датчика.

Таблица 24 – Примеры для объектного окна аналогового датчика

Выражение	Значение
Card.Marka	Марка датчика
ЗначениеСигнала.APar.ParamMin	Минимум диапазона параметра «Значение сигнала»
Remicont.Version	Версия Ремиконта, в котором находится соответствующий алгоблок ФАО.

Перечень доступных полей Базы данных приведен в разделе 7.

5.4.4.4 Использование библиотечных функций

Кроме функций, описанных прямо в тексте программы, можно использовать несколько десятков библиотечных функций. Примеры:

```
{ Вычисление наименьшего из значений параметров двух объектов }
X := Min (Объект ('M1').ЗначениеСигнала, Объект ('M2').ЗначениеСигнала);

{ Запись в консоль значения параметра в виде строки
с 3-мя знаками после запятой }
Write (Строка (Объект ('M1').ЗначениеСигнала, 3));
```

Библиотечные функции подробно описаны в 7.2.

Набор функций можно расширить, подключив дополнительные DLL, об этом см. в 5.4.6.

5.4.4.5 Управление элементами ActiveX

Каждый класс ActiveX предоставляет набор свойств и методов для работы с объектами этого класса. С помощью встроенной программы можно получать и изменять свойства объекта ActiveX и вызывать его методы. Обращение к свойствам и методам производится по имени.

Действует ограничение: тип значения свойства, а также типы параметров метода должны быть числовыми или строковыми.

5.4.5 Как создать программу

Для написания программы следует выбрать требуемый элемент изображения и нажать кнопку «Программа» в панели инструментов или выбрать строку «Программа» локального меню элемента. Открывшееся окно редактирования программы имеет следующие области:

- поле редактирования;
- меню;
- панель инструментов;
- панель со списками;
- панель сообщений.

Инд. №	Подп. и дата
Взаим. инв. №	
Инд. № подл.	

В поле редактирования можно вводить любой текст, используя те же приемы, что и в стандартном редакторе **Блокнот (Notepad)**. С помощью меню или панели инструментов можно также:

- откомпилировать текст (при наличии в тексте ошибки она выводится в панель сообщений а в поле редактирования выделяется первый ошибочный фрагмент; если ошибки нет - в панели сообщений появляется текст «ОК»);
- закрыть окно с сохранением текста (при этом автоматически вызывается компиляция, при наличии ошибки окно не закрывается);
- закрыть окно без сохранения текста;
- вставить в текст одну из процедур-обработчиков сообщений.

Панель со списками содержит два списка: список свойств (верхний) и список значений (нижний). Содержание нижнего списка зависит от выбранной строки в верхнем списке и представляет список возможных значений для выбранного свойства. Некоторые списки значений зависят более чем от одного выбранного свойства, например, список параметров объекта зависит от предварительно выбранного типа объекта, а список марок - от выбранных типа объекта и узла. Элементы обоих списков можно перетаскивать левой кнопкой мыши в поле редактирования.

5.4.6 Подключение динамических библиотек

Из встроенной программы можно вызывать функции динамических библиотек (DLL). Для этого функция должна иметь специальный прототип. Кроме того, функция и содержащая ее DLL должны быть зарегистрированы в файле **program.ini**. Этот файл расположен в подпапке **Config** той папки, в которую был установлен Квинт.

5.4.6.1 Требования к вызываемой функции

К вызываемой функции имеются два требования:

- 1 Все параметры и возвращаемое значение функции должны иметь тип `double`.
- 2 Функция должна иметь тип вызова `cdecl`.

Таким образом, функция должна быть объявлена примерно следующим образом (в случае двух параметров):

```
double __cdecl MyFunc(double x, double y); // C или C++
function MyFunc(x, y: double): double; cdecl; // Pascal или Delphi
```

Чтобы избежать декорирования экспортируемых имен, рекомендуется добавить директиву `extern "C"`.

5.4.6.2 Регистрация DLL в файле program.ini

В секции [Libraries] следует добавить строку вида

`DllName=DllPath`, где

- **DllName** - задаваемый пользователем псевдоним DLL,
- **DllPath** - имя файла DLL.

Если DLL находится в папке **Bin** Квинта или в одной из папок, перечисленных в переменной среды **Path**, то имя файла DLL можно указывать без полного пути.

В секции [Functions] следует добавить строку вида

`FuncName=DllName ExportedName ParamCount`, где

- **FuncName** - задаваемый пользователем псевдоним функции,
- **DllName** - псевдоним DLL,
- **ExportedName** - реальное экспортируемое имя функции,
- **ParamCount** - кол-во параметров функции.

5.4.6.3 Вызов функции из встроенной программы

Функция вызывается так:

```
FuncName(Param1, Param2, ..., ParamN);
```

, где

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

- **FuncName** - псевдоним функции,
- **Param1 ... ParamN** - значения передаваемых параметров.

Например: Библиотека myutils.dll содержит функцию

```
extern "C" double __declspec(dllexport) __cdecl Average(double x, double y)
{
    return (x + y) / 2;
}
```

Регистрируем ее следующим образом:

```
[Libraries]
myutils=C:\MyProjects\myutils.dll
```

```
[Functions]
Average=myutils Average 2
```

Вызываем так:

```
Average(Объект('A1').ЗначениеСигнала, Объект('A2').ЗначениеСигнала);
```

5.4.7 Отладка программ

Для отладки встроенной программы можно выдавать диагностические сообщения в окно Квинтегратора. Для генерации таких сообщений следует в тексте программы использовать команду `Write`. Пусть `x` - некоторая переменная и мы хотим распечатать ее значение. Вставим в текст программы команду

```
Write('Значение x = ', x);
```

Тогда при `x = 5` в окне **Квинтегратор** будет выведен текст:

```
Значение x = 5
```

5.5 Отладка динамики


5.5.1 Режим обзора


Каждое открытое в Графите изображение находится либо в режиме редактирования, либо в режиме обзора. Отличия этих режимов сведены в таблицу 25.

Таблица 25 – Отличия режимов редактирования и обзора

	Режим редактирования	Режим обзора
Правка изображения	Допускается	Не допускается
Аниматоры, рецепторы, программы	Не выполняются	Выполняются

В режиме обзора изображение ведет себя так же, как и в Операторской станции, за исключением моментов, связанных с общей компоновкой экрана.

Кнопка  переводит изображение в активном окне из режима редактирования в режим обзора. Если редактируемое изображение – типовое, то при переводе в обзор нужно указать марку. Это связано с тем, что обзор всегда работает с конкретными, а не с типовыми параметрами.

Кнопка  возвращает изображение в режим редактирования. Если изображение было открыто в результате срабатывания рецептора или программы, то оно сразу открывается в режиме обзора, и перевести его в режим редактирования нельзя.

5.5.2 Эмулятор

5.5.2.1 Назначение эмулятора

Как уже упоминалось, Графит можно запустить в режиме **С эмулятором** или **С контроллером**. Если Графит запущен **С эмулятором**, то в режиме обзора, вместо запроса значений параметров из их настоящего источника, Графит получает их из эмулятора.

Эмулятор является программной компонентой. Он либо формирует значения параметров случайным образом, либо дает возможность пользователю задавать их интерактивно в специальном окне.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Эмулятор предназначен, главным образом, для проверки адекватности заданной в Графите анимации. Эмулятор не моделирует технологические объекты и не является средством отладки технологических программ. Для этой цели следует использовать Мезон.

5.5.2.2 Работа с эмулятором

Если изображение анимируется хотя бы одним параметром, то при переходе в режим обзора создается окно эмулятора. Это окно появляется в свернутом виде в панели задач Windows и имеет заголовок **Эмулятор**. Для открытия окна щелкните мышью по этому заголовку.

Назначение элементов окна эмулятора представлено на рисунке 29. В окне можно выбирать параметры, участвующие в анимации изображения, и изменять их свойства: значение, предустановку, статус, качество. При изменении этих свойств изображение в окне Графита будет соответствующим образом изменяться. В случае сложной логики анимации рекомендуется сравнивать полученный эффект с ожидаемым при различных комбинациях значений параметров, задаваемых в окне эмулятора.

Кнопка **Настройки...** открывает дополнительное диалоговое окно. Если в этом окне установлен флажок **Включить**, эмулятор будет изменять значения параметров случайным образом. Для каждого типа параметров можно указать, следует ли автоматически изменять эти параметры (отдельно для управляемых и неуправляемых параметров), а также средний интервал времени изменения и задержку изменения при управлении параметром в Графите.

В окне настроек можно также включить случайную генерацию ошибок, которые будут отображаться в списках сигнализации. Введенные настройки начинают действовать при нажатии кнопок **ОК** или **Применить** и автоматически сохраняются в реестре. Значения настроек зависят от пользователя и компьютера, но не зависят от проекта.

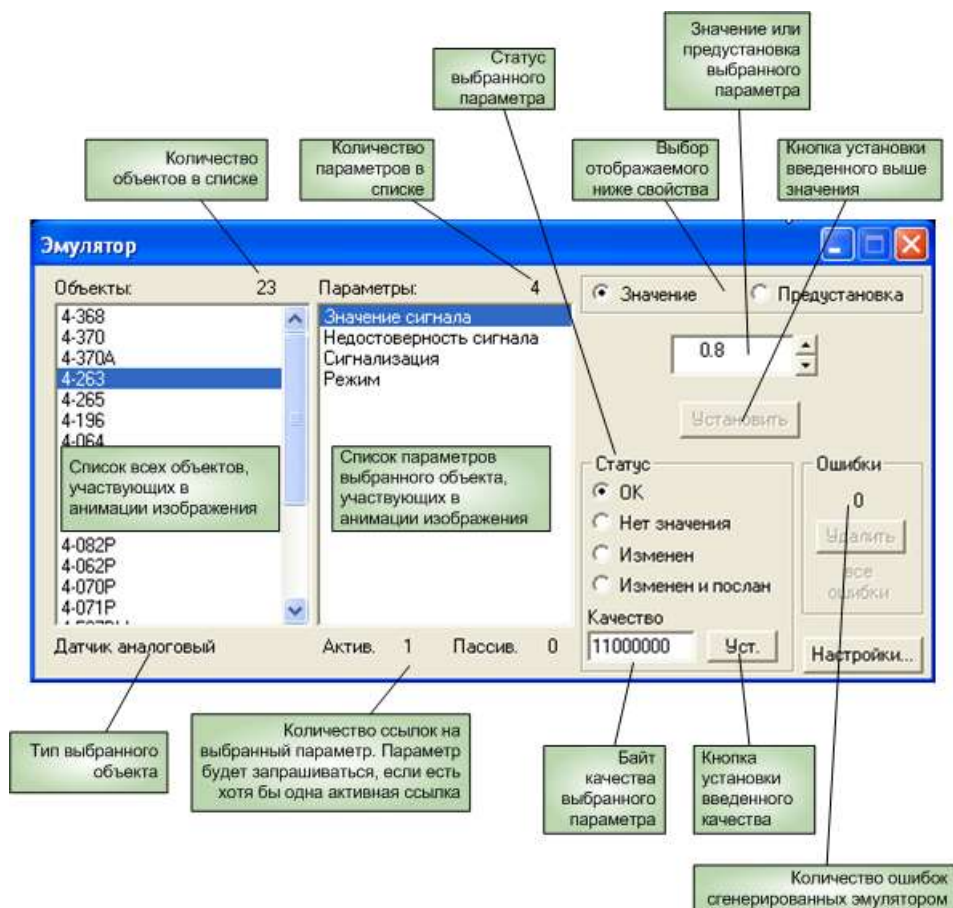


Рисунок 29 – Окно Эмулятора

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

6 Дополнительные возможности

6.1 Настройки

Настройка параметров Графита производится:

- из пункта **Вид** меню главного окна. Эти настройки описаны в 7.2.1;
- из приложения **Настройка – Параметры**, узел **Графит**. Пояснения к этим настройкам приводятся в окне приложения.

Примечание - Настройки, сделанные из приложения **Настройка – Параметры**, вступают в силу только после перезапуска Графита.

Более сложными являются настройки палитры и стилей текста, описанные ниже.

6.1.1 Настройка палитры

6.1.1.1 Цвета

В Графите широко используется понятие цвета. Пользователь задает цвет заполнения элемента, а также цвет его контура, фона, текста и т.д. В данном разделе мы будем для краткости употреблять слово **цвет**, подразумевая под ним цвет заполнения или другой из подобных атрибутов.

При задании цвета пользователь выбирает один из цветов в палитре. Палитру можно представлять себе как таблицу из **N** строк (рисунок 30), в которой каждому номер сопоставлен определенный цвет:

0	
1	
2	
3	
...	...
N - 1	

Рисунок 30 – Пример создания таблицы палитры

Каждый элемент хранит свой цвет как число из левого столбца этой таблицы. Элемент выводится на экран с цветом из соответствующей ячейки правого столбца. Если поменять цвет в правой ячейке, то изменится вид не только этого элемента, но и всех элементов, имеющих цвет с тем же номером. Более того, эти изменения коснутся всех изображений, входящих в данный проект. Тем самым обеспечивается единство цветовой гаммы для всех изображений проекта.

Палитра хранится в базе данных проекта. При создании нового проекта количество цветов **N** = 20, и палитра совпадает с системной палитрой Windows. В процессе проектирования количество цветов и их значения могут быть изменены.

6.1.1.2 Осцилляторы

Для заполнения, контура и других атрибутов элемента можно задать не только цвет, но и частоту мигания. Частота мигания указывается не произвольным образом, а выбирается из списка. Список содержит элемент 0 Гц, означающий ровное свечение, и еще **M** строк, соответствующих разным **осцилляторам**. Каждый осциллятор определяет частоту мигания всех связанных с ним элементов. Кроме того, для осциллятора определяется **альтернативный цвет**. При мигании цвет элемента периодически заменяется альтернативным цветом осциллятора.

Осцилляторы, также как и набор цветов, хранятся в базе данных проекта. По умолчанию, количество осцилляторов **M** = 2 и альтернативный цвет у них – нулевой. В процессе проектирования эти настройки могут быть изменены.

6.1.1.3 Диалог настройки

Для настройки таблицы цветов и осцилляторов вызовите в Графите пункт меню **Вид / Настройка палитры**. В открывшемся диалоговом окне можно:

- менять количество цветов;
- изменять значение выбранного цвета;

Инд. № подл.	
Подп. и дата	
Взаим. инв. №	
Инд. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

- давать название выбранному цвету;
- добавлять и удалять осцилляторы;
- изменять частоту и альтернативный цвет выбранного осциллятора;
- давать название выбранному осциллятору;

В диалоге указывается не частота в герцах, а полупериод мигания в миллисекундах. Эти величины связаны соотношением **Частота * Полупериод = 500**.

Названия цветов и осцилляторов можно использовать во встроенной программе, описанной в 7.3.4.3.

6.1.1.4 Особенности реализации

В Квинте версии 5, для ускорения прорисовки экрана, мигание было реализовано системными средствами методом переключения цветовой палитры. Microsoft Windows поддерживает такую возможность только в режиме 256 цветов. Квинт-6 рассчитан на современные видеоадаптеры, и мигание элементов в нем осуществляется с помощью обычной перерисовки.

Основные отличия Квинта по этим вопросам приведены в таблице 26.

Таблица 26 – Отличия прорисовки экрана

	Квинт 5	Квинт-6
Настройка видеоадаптера по глубине цвета	256 цветов	Любая (рекомендуется TrueColor)
Ограничение по количеству цветов N и количеству осцилляторов M	$N * (M + 1) < \text{или} = 118$	$N < \text{или} = 64$ (связано только с размером окна палитры в диалоговых окнах). M – любое.
Мигание в Графите	В любом режиме	Только в режиме обзора
Перекрытие мигающего элемента немигающим	Всегда возможно	Элементы нужно объединять и настраивать объединение (3.2.2)


6.1.2 Настройка стилей текста

Для вывода текстов Графит позволяет использовать любые шрифты, установленные в Windows. Чтобы упорядочить многообразие используемых в проекте шрифтов и упростить их настройку, в Графите введено понятие **стиля** текста.

Каждый стиль имеет:



- внутренний номер;
- имя;
- описание шрифта (название, размер, начертание).

В проекте хранится единая для всех изображений таблица стилей. Каждый элемент изображения, содержащий строки (Текст, Шкала, Реквизит и др.), хранит номер стиля для своего текста. При изменении настроек стиля изменяется вид всех элементов, связанных с этим стилем, на всех изображениях проекта.

Для настройки таблицы стилей нужно вызвать кнопкой  диалоговое окно настройки текста. В зоне **Стиль** этого окна находится список имен имеющихся стилей ряд кнопок:

- **Новый** – предлагает ввести имя нового стиля, затем открывает стандартное окно выбора шрифта. После нажатия кнопки **ОК**, стиль добавляется в таблицу.
- **Изменить** – для выбранного в списке стиля вызывает стандартное окно настройки шрифта. Изменение настроек шрифта влияет на все изображения;
- **Переименовать** – позволяет изменить имя стиля, выбранного в списке. Изменение имени влияет только на представление стиля в списке;
- **Удалить** – удаляет выбранный в списке стиль.

Примечания

- 1 После удаления стиля различные изображения проекта могут стать некорректными (см. 6.4).
- 3 Рекомендуется при задании стиля выбирать только шрифты, помеченные символами  (TrueType) или  (OpenType). Такие шрифты являются векторными и лучше масштабируются.

Подп. и дата	
Инв. №	
Взаим. инв. №	
Подп. и дата	
Инв. № подл.	

6.1.3 Копирование настроек При создании нового проекта нерационально задавать все настройки «с нуля»: их можно скопировать из существующего проекта. В **Мастере добавления проектов** укажите параметр **Выбрать разделы для копирования** и далее выберите разделы:

- **Палитра и стили текста.** При этом из исходного проекта будут скопированы настройки палитры и таблица стилей;
- **Настройки.** При этом из исходного проекта будут скопированы настройки, задаваемые в приложении **Настройка – Параметры**. Настройки, задаваемые в меню **Вид** не копируются, при необходимости их следует указать заново.

Можно также копировать настройки из одного проекта в другой с помощью приложения **Слияние БД**.

6.2 Печать изображений

6.2.1 Печать

Чтобы напечатать активное изображение, вызовите пункт меню **Файл / Печать**. В открывшемся диалоговом окне **Печать** выберите принтер и количество копий, затем нажмите кнопку **ОК**.

Если в диалоговом окне **Печать** выбрать параметр **Выделенный фрагмент (Selection)**, то будет печататься не все изображение, а только выделенные его элементы.

Примечание - Поддерживается печать только векторных изображений, для печати картинок используйте внешние графические редакторы.

6.2.2 Настройка параметров страницы

Пункт меню **Файл – Параметры страницы** вызывает одноименное диалоговое окно, в котором можно указать размеры и ориентацию бумаги. Размер полей в данной версии не поддерживается. Настройка действует на все печатаемые изображения.

При печати масштаб автоматически подгоняется так, чтобы изображение целиком поместилось на листе бумаги. Размер изображения определяется по маркеру размера, а не по фактическому размеру элементов.

6.2.3 Печать фона

По умолчанию, фон изображения не выводится на печать. Если это необходимо, укажите параметр **Графит - Выводить фон изображения на печать** в приложении **Настройка – Параметры**.

6.3 Импорт и экспорт изображений

Под импортом и экспортом понимается перенос каких-либо проектных данных вне рамок одного проекта Квинта.

6.3.1 Копирование изображения из одного проекта в другой

6.3.1.1 Копирование одного изображения

Чтобы скопировать одно изображение из проекта **А** в проект **В** необходимо последовательно выполнить следующие действия:

- 1 Выберите в окне **КВИНТегратор** проект **А** и запустите **Графит**.
- 4 Откройте окно выбора изображения, выберите требуемое изображение и нажмите кнопку **Копировать**. Изображение будет скопировано в буфер обмена.
- 5 Закройте **Графит**, выберите в окне **КВИНТегратор** проект **В** и запустите **Графит** снова.
- 6 Откройте окно выбора изображения, выберите раздел (узел или тип), в который должно быть скопировано изображение и нажмите кнопку **Вставить**. Изображение из буфера обмена будет добавлено в выбранный раздел.
- 7 Откройте добавленное изображение и проверьте его.

6.3.1.2 Копирование нескольких изображений

Если нужно скопировать несколько изображений, можно многократно повторить описанную выше процедуру, но удобнее и быстрее последовательно выполнить следующие действия:

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						87

- 1 Выберите в окне **КВИНТегратор** проект **А** и запустите Графит.
- 2 Запустите еще один экземпляр **КВИНТегратор**, выберите в нем проект **В** и запустите для него другой экземпляр Графита. В заголовке главного окна Графита отображается имя проекта – это позволит различить экземпляры Графита.
- 3 Производите операции копирования и вставки, переключаясь между окнами двух Графитов с помощью клавиш **Alt+Tab**.

6.3.1.3 Ограничения при копировании

При копировании изображений существуют три ограничения:

- 1 Изображения можно разделить на 3 группы: типовые, нетиповые, картинки (см. 2.1.1). Копировать изображения можно только в пределах группы, например, нельзя картинку скопировать как мнемосхему, но можно объектное окно скопировать как мнемосимвол.
- 4 Типовое изображение (мнемосимвол или объектное окно) можно добавить только в список типовых изображений **того же оперативного типа**.
- 5 Версия проекта-источника не должна быть выше версии проекта-приемника, иначе скопированное изображение может не открыться.

6.3.1.4 Проблемы при копировании

Изображение, как документ Графита, не является «самодостаточным»: оно содержит ссылки на другие изображения, марки и прочие объекты проекта. При переносе в другой проект эти ссылки, с большой долей вероятности, станут некорректными.

Приведем пример:

Пусть мнемосхема **М** копируется из проекта **А** в проект **В**. Предположим, что на нее была вставлена картинка **Пускатель**, имеющая в проекте идентификатор **ID**. Мнемосхема хранит в своих данных этот **ID**. После копирования мнемосхемы возможны 3 ситуации:

- 1 В проекте **В** есть аналогичная картинка **Пускатель** с тем же идентификатором **ID**. В этом случае никаких проблем нет.
- 2 В проекте **В** есть аналогичная картинка **Пускатель**, но она имеет другой идентификатор. Тогда в скопированной мнемосхеме нужно скорректировать ссылку, вызвав для вставки пункт меню **Выбрать картинку**.
- 3 В проекте **В** нет нужной картинки. В этом случае нужно картинку **Пускатель** тоже скопировать из проекта **А** в проект **В**. Если в проекте **В** не было изображения с идентификатором **ID**, то копия картинки получит тот же идентификатор **ID** и мнемосхема исправится автоматически. Если же в проекте **В** под идентификатором **ID** было другое изображение, скопированная картинка получит новый идентификатор, и нужно будет скорректировать ссылку аналогично пункту 2.

Другим источником «порчи» изображения при переносе может явиться несогласованность цветовых палитр в проектах **А** и **В**. В атрибутах элементов изображения хранится номер цвета в палитре. В другом проекте под тем же номером в палитре может находиться другой цвет, или вообще отсутствовать цвет. В результате, после вставки может потребоваться «перекрасить» отдельные элементы.

После вставки необходимо произвести проверку вставленного изображения и исправить некорректные ссылки. Подробнее об этом см. в 6.4.

6.3.2 Замена содержимого изображения

Иногда требуется, чтобы при копировании изображения из проекта **А** в проект **В** не создавалось новое изображение, а корректировалось старое. Например, если нужно подменить в проекте **В** некоторое объектное окно на окно из проекта **А**, процедура, описанная в предыдущем разделе, может оказаться неудобной, поскольку потом придется заменять все рецепторы вызова этого окна в других изображениях.

Для замены содержимого окна необходимо последовательно выполнить следующие действия:

- 1 Выберите в Квинтеграторе проект **А** и запустите Графит.
- 2 Откройте исходное изображение, вызовите команды **Правка / Выделить все** и **Правка / Копировать**. Содержимое изображения будет скопировано в буфер обмена.
- 3 Закройте Графит, выберите в Квинтеграторе проект **В** и запустите Графит снова.
- 4 Откройте изображение, которое нужно подменить, вызовите команды **Правка / Выделить все**, **Правка / Удалить** и **Правка / Вставить**.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

5 Если нужно, поправьте размер изображения и проверьте его.

Примечание - При выполнении указанных действий не копируются свойства изображения в целом: название, размер, цвет фона и др. Также не копируются те аниматоры, рецепторы и встроенная программа, которые относятся к изображению в целом.

Описанную процедуру можно применять не ко всему изображению, а к его фрагменту. Для этого вместо операции **Выделить все** выделите требуемые элементы изображения.

6.3.3 Сохранение фрагмента изображения в виде файла

Содержимое изображения или его части можно скопировать не только в буфер обмена, но и в файл (например, при необходимости скопировать данные на другой компьютер). Для этого откройте изображение, выделите нужные элементы и вызовите команду **Файл / Экспорт**. Выберите тип файла «gr2» и введите имя файла.

Для вставки ранее созданного файла «gr2» в другое изображение в том же или в другом проекте используйте команду **Файл / Импорт**. Данные из файла будут добавлены к активному изображению в сгруппированном виде.



Примечание - Файл gr2 не содержит свойств изображения в целом, таких как: название, размер, цвет фона и др. Также не в нем не сохраняются те аниматоры, рецепторы и встроенная программа, которые относятся к изображению в целом.

6.3.4 Копирование рисунков из других программ

Копирование растровых изображений из других графических редакторов описано в 2.7.

В этом разделе описано копирование фрагментов векторных изображений в Графит из других редакторов векторной графики. Это средство не является необходимым для работы в Графите, но к нему удобно прибегать в следующих случаях:

- другой редактор предоставляет некоторый инструмент рисования, отсутствующий в Графите. Например, пусть требуется нарисовать правильный 5-угольник (у которого все стороны равны). В Графите пришлось бы рассчитывать и вводить его координаты (3.1.5). Намного проще нарисовать многоугольник в Microsoft Word с помощью инструмента **Автофигуры**, а затем скопировать его в Графит;
- имеется готовая схема, выполненная в другом векторном редакторе, которую нужно внедрить в Графит. Если нужно сделать эту схему статическим фоновым рисунком на мнемосхеме, можно перенести схему как растровое изображение (картинку), это обеспечит точный перенос без каких-либо искажений. Если же требуется более тесно интегрировать схему в Графит, наложить анимацию и управление, то следует копировать схему как векторное изображение.

Для копирования откройте исходный рисунок редактором, в котором рисунок был создан, и скопируйте его в буфер обмена. Затем откройте в Графите изображение, в которое должен быть вставлен рисунок, и выполните команду **Правка / Вставить**. Рисунок будет вставлен в виде нескольких сгруппированных элементов. Если элементы нуждаются в коррекции, выполните операцию  (**Внутри объединения**) или  (**Разъединить**).

Альтернативой использования буфера обмена является сохранение рисунков в виде метафайлов с расширениями WMF или EMF и вставка их в Графит с помощью команды **Файл / Импорт**. Такой подход позволяет перенести рисунок с одного компьютера на другой.

Примечание - При копировании рисунков из других программ, полученная в Графите копия может существенно отличаться от оригинала по ряду свойств. Это является следствием различных ограничений на используемые графические примитивы в Графите и в оригинальном редакторе рисунка.

6.3.5 Экспорт рисунков в другие программы

Векторные изображения Графита можно переносить в другие приложения, поддерживающие векторную графику, например, в Microsoft Word или Microsoft Power Point. Для этого откройте изображение в Графите, выделите все элементы или только требуемые, скопируйте их в буфер обмена. Затем откройте документ в другом редакторе и вставьте изображение из буфера обмена.

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

Если при этом нужно перенести изображение на другой компьютер, сохраните выделенные элементы в метафайле с расширением EMF с помощью команды **Файл / Экспорт**.

6.4 Проверка изображений

6.4.1 Для чего нужна проверка

В процессе редактирования изображения Графит «следит» за тем, чтобы вносимые пользователем изменения не нарушали корректность изображения. Но почти все изображения содержат ссылки на различные объекты, внешние по отношению к изображению: марки, файлы и др. Если эти объекты впоследствии удалить, изображение станет **некорректным**. Это означает, что в оперативном режиме такое изображение либо вообще не откроется, либо будет отображать неполную информацию.

Примечание - При завершении проектных и наладочных работ необходимо убедиться в **корректности всех изображений** проекта.

Процедура проверки позволяет выявить некорректные изображения и локализовать ошибки в них.


6.4.2 Примеры ошибок

В таблице 27 приведены некоторые типичные ошибки и методы их исправления.

Таблица 27 – Ошибки и методы их исправления

Текст ошибки	Пояснения	Как исправить
Отсутствует изображение вызываемое рецептором	Некоторый элемент имеет рецептор открытия окна с другим изображением. Это другое изображение отсутствует в проекте. (Оно могло быть удалено, либо рецептор мог быть скопирован из другого проекта).	Удалить этот рецептор, либо вызвать для него пункт контекстного меню Изображение и указать другое изображение.
Объект мнемосимвола не входит в один из срезов, в которые входит эта мнемосхема	В проекте имеется срез X. Мнемосхема входит в этот срез, а марка M, связанная с данным мнемосимволом – не входит. Если для компьютера установлен рабочий срез X, то при открытии этой мнемосхемы на операторской станции возникнет ошибка, потому что марка M будет отсутствовать в DAT-файле.	Включить марку M в срез X или удалить ее мнемосимвол с этой мнемосхемы.
Отсутствует стиль текста с номером N	Для элемента, содержащего текст, указан несуществующий стиль текста. (Стиль мог быть удален, либо элемент мог быть скопирован из другого проекта).	Вызвать окно задания атрибутов и указать в нем другой стиль текста.
Ошибка во встроенной программе	Встроенная программа также может содержать ссылки на марки и другие внешние объекты. Ошибка может быть связана с отсутствием этих объектов.	Открыть окно встроенной программы для выделенного элемента, нажать в нем кнопку Проверка . Исправить ошибку в тексте.
qAXSupp: GUID = '{...}' Windows(-2147221164): Class not registered	Изображение содержит элемент ActiveX, не зарегистрированный в системе. При использовании ActiveX необходимо, чтобы он был установлен на всех компьютерах, на которых запускаются Графит или операторская станция.	Проинсталлируйте требуемый ActiveX или удалите его из изображения.

6.4.3 Проверка одного изображения

Чтобы проверить изображение, нужно открыть его в Графите и нажать кнопку  или вызвать пункт меню **Сервис / Проверить изображение**.

Если изображение корректно, будет выдано сообщение с текстом **Ошибок не обнаружено**.

Если изображение некорректно, при нахождении первой ошибки будет выдано сообщение с ее описанием. При этом элемент, содержащий ошибку, будет выделен маркером, а в списке элементов будет выделена соответствующая строка. Если ошибка связана с аниматором или рецептором, будет выделена соответствующая строка в списке аниматоров или рецепторов для данного элемента.

После исправления найденной ошибки следует снова вызвать проверку и продолжать эту процедуру до тех пор, пока изображение не станет корректным. Исправленное изображение необходимо сохранить.

Инва. № подл.	Подп. и дата	Взаим. инв. №	Инва. №	Подп. и дата
---------------	--------------	---------------	---------	--------------

6.4.4 Проверка всех изображений

Для проверки всех изображений в Графите нажмите кнопку  или вызовите пункт меню **Сервис / Проверить все**.


Примечание - Для проверки всех изображений рекомендуется запускать **Тест изображений** из компоненты **Администрирование / Диагностика проекта**. Этот тест выявляет те же ошибки, что и Графит, но позволяет вывести сразу все ошибки, в то время как проверка в Графите останавливается на первой найденной ошибке.

Открывшееся окно **Проверка всех изображений** содержит информацию, приведенную в таблице 28.

Таблица 28 – Содержимое окна Проверка всех изображений

Зона	Элемент	Назначение
Панель статистики	Всего изображений	Общее количество изображений в проекте
	Обработано	Сколько изображений было обработано, пока окно проверки было открыто
	Обновлено	Сколько изображений было автоматически сохранено
	Объем на диске	Объем памяти в БД проекта (и в DAT-файле), занимаемый всеми протестированными изображениями
	Объем в ОЗУ	Объем оперативной памяти, занимаемый всеми протестированными изображениями
Флажки настройки	Проверка	Для каждого изображения будет выполняться проверка
	Обновление	Если изображение создано в старой версии Графита, его формат будет автоматически обновлен и изображение будет сохранено. В дальнейшем это может уменьшить время открытия изображения.
	Статистика	Для каждого изображения будет вычисляться объем занимаемой им дисковой и оперативной памяти.
	Привязка	Выдавать ошибку, если изображение использует марки, не привязанные к алгоблокам.
	Коррекция	Устанавливается автоматически при выполнении процедуры коррекции идентификаторов БД.
Кнопки управления	Пуск	Запускает обработку изображений, начиная с выделенного изображения в списке. Ранее обработанные изображения пропускаются. Кнопка активна, только если все изображения в Графите закрыты.
	Стоп	Приостанавливает процедуру обработки
	Закрыть	Закрывает окно. При закрытии окна утрачивается информация об обработанных окнах, поэтому рекомендуется не закрывать окно, пока остаются неисправленные ошибки.
Список изображений	Содержит список всех изображений проекта. Обработанное изображение помечается синей галочкой, если оно корректно, или красным кружком, если в нем есть ошибки.	

Рекомендуемая последовательность действий при проверке всех изображений:

- 1 Закройте все окна изображений в Графите (окна со списками элементов, аниматоров и рецепторов рекомендуется открыть).
- 2 Откройте окно **Проверка всех изображений**. По умолчанию будет установлен флажок **Проверка**.
- 3 Нажмите кнопку **Пуск**.
- 4 Если все элементы списка изображений помечены синей галочкой, то процедуру проверки можно считать успешной.
- 5 При обнаружении ошибки в одном из изображений это изображение будет открыто и в нем будет выделен элемент, содержащий ошибку.
- 6 Исправьте ошибку, не закрывая окна **Проверка всех изображений**, после чего нажмите кнопку .
- 7 Пока выдается сообщение об ошибке, повторяйте пункт 6.
- 8 Если ошибок не найдено, сохраните изображение и закройте его.
- 9 Перейдите к пункту 3.

Если ошибку нецелесообразно исправлять на данном этапе работ, ее можно пропустить. Для этого закройте окно изображения, открывшееся при выполнении пункта 5, выделите в списке изображений следующий элемент и нажмите кнопку **Пуск**.

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Подп. и дата

7 Справочник

В этом разделе приведена различная справочная информация по меню, панели инструментов Графита и функциям встроенной программы.

7.1 Описание пользовательского интерфейса

Файл Правка Вид Преобразования Динамика Сервис Окна

7.1.1 Главное менюФайл

Элемент главного меню **Файл** содержит следующие пункты:

- **Открыть.** Содержит подменю, в котором можно выбрать тип открываемого изображения:
 - а) **Мнемосхема.** Открывает диалоговое окно **Выбор мнемосхемы**;
 - б) **Объектное окно.** Открывает диалоговое окно **Выбор типового изображения**;
 - в) **Мнемосимвол.** Открывает диалоговое окно **Выбор типового изображения**;
 - г) **Картинка.** Открывает диалоговое окно **Выбор фигуры**;
 - д) **Фигура.** Открывает диалоговое окно **Выбор фигуры**;
- **Заккрыть.** Закрывает активное изображение;
- **Сохранить.** Сохраняет активное изображение;
- **Сохранить все.** Сохраняет все открытые изображения;
- **Автосохранение.** Открывает диалоговое окно для задания параметров автосохранения – флаг возможности автосохранения, период автосохранения, флаг вывода запроса перед автосохранением;
- **Печать.** Выводит активное изображение на печать;
- **Параметры страницы.** Открывает диалоговое окно установки параметров страницы (подробнее см. 6.2);
- **Импорт.** Открывает диалоговое окно выбора файла для импорта изображения;
- **Экспорт.** Открывает диалоговое окно экспорта в файл активного изображения;
- **Параметры окна.** Открывает диалоговое окно **Параметры изображения**. С его помощью можно задать размеры и цвет фона активного изображения;
- **Информация.** Выводит информацию об активном изображении:
 - а) идентификатор изображения;
 - б) имя изображения, тип изображения;
 - в) версия программного обеспечения;
 - г) объем, занимаемый изображением на диске;
 - д) объем, занимаемый изображением в оперативной памяти;
- **Выход.** Закрывает приложение. Если открытые изображения не были сохранены, выдается запрос на их сохранение. Горячие клавиши – **Alt+F4**.

7.1.1.2 Правка

Элемент главного меню **Правка** содержит следующие пункты:

- **Отменить.** Отменяет последнее действие пользователя. Количество уровней отмены задается в общих настройках Квинтегратора. Горячие клавиши – **Alt+BackSpace**;
- **Восстановить.** Восстанавливает вид изображения, предшествовавший отмене. Горячие клавиши – **Alt+Shift+BackSpace**;
- **Вырезать.** Помещает выделенные элементы изображения в буфер обмена и удаляет их из изображения. Горячие клавиши – **Shift+Del**;
- **Копировать.** Помещает выделенные элементы изображения в буфер обмена. Горячие клавиши – **Ctrl+Ins**;
- **Вставить.** Вставляет в изображение элементы, находящиеся в буфере обмена. Горячие клавиши – **Shift+Ins**;
- **Удалить.** Удаляет выделенные элементы из изображения. Горячие клавиши – **Del**;
- **Удвоить.** Вставляет в изображение копию выделенных элементов. Горячие клавиши – **Ctrl+D**;
- **Выделить все.** Выделяет все элементы изображения;

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

- **Выделить по атрибуту.** Открывает диалоговое окно **Выбор по атрибуту**, в котором перечислены все атрибуты элементов изображения. В активном изображении будут выделены элементы, у которых выбранный атрибут имеет выбранное значение;
- **Изменить примитив.** Открывает специфические диалоги настроек выбранного элемента. Если таких диалогов нет, то этот пункт неактивен. Горячие клавиши – **Ctrl+P**.

7.1.1.3 Вид

Элемент главного меню **Вид** содержит следующие пункты:

- **Увеличить.** Увеличивает масштаб активного изображения вдвое;
- **Уменьшить.** Уменьшает масштаб активного изображения вдвое;
- **Исходный размер.** Возвращает исходный масштаб активного изображения;
- **Сохранить.** Сохраняет текущую конфигурацию окон в качестве исходной для операторской станции. Эта функция описана в 4.4;
- **Примитивы.** Если у пункта меню **Примитивы** не установлен флаг присутствия на экране окна **Примитивы** - открывает соответствующее окно, в противном случае закрывает его. Окно **Примитивы** содержит список всех элементов изображения, позволяет изменять их порядок и производить некоторые групповые операции. Например, объединить выделенные элементы, удалить их, удвоить и т.д. Выделение элементов в списке осуществляется либо одновременным нажатием левой кнопки мыши и клавиши **Ctrl** (элементы выделяются по одному), либо сначала левой кнопкой мыши выделите первый элемент группы, а затем одновременным нажатием левой кнопки мыши и клавиши **Shift** – последний элемент группы. Выделенные в списке элементы выделяются и на самом изображении, и наоборот. Для каждого элемента списка можно вызвать его контекстное меню;
- **Аниматоры.** Если у пункта меню **Аниматоры** не установлен флаг присутствия на экране окна **Аниматоры** - открывает соответствующее окно, в противном случае закрывает его. Окно **Аниматоры** содержит список существующих аниматоров элемента и позволяет добавлять новые;
- **Рецепторы.** Если у пункта меню **Рецепторы** не установлен флаг присутствия на экране окна **Рецепторы** - открывает соответствующее окно, в противном случае закрывает его. Окно **Рецепторы** содержит список существующих рецепторов элемента и позволяет добавлять новые;
- **Вид мнемосхемы.** Если у пункта меню **Вид мнемосхемы** не установлен флаг присутствия на экране окна **Вид мнемосхемы** - открывает соответствующее окно, в противном случае закрывает его. Окно **Вид мнемосхемы** позволяет изменять масштаб изображения;
- **Редактирование.** Открывает или закрывает панель инструментов для редактирования изображения. Если панель открыта, у пункта меню взводится флаг присутствия ее на экране;
- **Инструменты.** Открывает или закрывает панель инструментов для добавления элементов изображения. Если панель открыта, у пункта меню взводится флаг присутствия ее на экране;
- **Палитра.** Открывает или закрывает панель палитры. Если панель открыта, у пункта меню взводится флаг присутствия ее на экране;
- **Сетка.** Открывает диалоговое окно **Свойства сетки**, которое позволяет задать интервал сетки по осям X и Y; смещение сетки относительно левого верхнего угла окна изображения по осям X и Y; установить флаги видимости сетки в окнах изображений и «прилипания» элементов изображения к сетке;
- **Настройка палитры.** Открывает диалоговое окно **Настройка палитры**, которое позволяет задать количество цветов в палитре, дополнить ее новыми цветами, добавить изменить или удалить осцилляторы. Эта функция подробнее описана в 6.1.1.
- **Цвет по умолчанию.** Открывает окно **Задание атрибутов**. Введенные значения атрибутов не присваиваются не какому-либо существующему элементу, а будут присваиваться всем новым создаваемым элементам;
- **Маркер.** Открывает окно **Задание атрибутов**, которое позволяет установить атрибуты раскраски маркера. Под маркером понимается рамка, возникающая вокруг выбранного элемента в оперативном режиме;
- **Линейки прокрутки.** Устанавливает признак автоматического появления линеек прокрутки в случае, если маркер размера изображения больше размера окна изображения. Точные размеры маркера изображения задаются в меню **Файл \ Параметры окна**;

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						93

- **Только контур.** Устанавливает для активного изображения признак рисования только контуров элементов;
- **Буксировка фигур.** Устанавливает признак буксировки фигур. Если признак буксировки установлен, то при перемещении элемента двигается его контур, в противном случае двигается прямоугольник, описанный вокруг элемента;
- **Перерисовать все.** Иницирует перерисовку всего изображения. Горячая клавиша – F9.

7.1.1.4 Преобразования

Элемент главного меню **Преобразования** содержит следующие пункты:

- **Атрибуты.** Открывает диалоговое окно задания атрибутов, которое позволяет изменить атрибуты выделенных элементов;
- **Выравнивание.** Открывает диалоговое окно **Выравнивание**. В нем задаются параметры выравнивания выделенных элементов друг относительно друга или относительно всего изображения. Если выделен только один элемент, то для него имеет смысл только выравнивание относительно всего изображения. С помощью данного диалога можно выровнять положение элементов по горизонтали и вертикали, а также размер элементов по горизонтали и вертикали;
- **Объединить.** Осуществляет операцию объединения нескольких элементов в один;
- **Разъединить.** Разъединяет объединение элементов;
- **Поворот.** Осуществляет поворот выделенного элемента на 90° против часовой стрелки;
- **Отражение.** Имеет два подпункта – по горизонтали и по вертикали. Осуществляет поворот выделенного элемента на 180° относительно горизонтальной или вертикальной оси;
- **Вперед.** Перемещает выделенный элемент на последнюю позицию порядка добавления элементов в изображение, если элемент не является последним;
- **Назад.** Перемещает выделенный элемент на первую позицию порядка добавления элементов в изображение, если элемент не является первым;
- **Вперед на один.** Перемещает выделенный элемент на одну позицию вперед в порядке добавления элементов в изображение, если элемент не является последним;
- **Назад на один.** Перемещает выделенный элемент на одну позицию назад в порядке добавления элементов в изображение, если элемент не является первым.

7.1.1.5 Динамика

Элемент главного меню **Динамика** содержит следующие пункты:

- **Удалить аниматоры.** Удаляет все аниматоры выделенного элемента;
- **Удалить рецепторы.** Удаляет все рецепторы выделенного элемента;
- **Старт.** Запускает для данного изображения эмулятор, в котором можно имитировать поведение динамических элементов. В режиме эмуляции редактирование изображения запрещено;
- **Стоп.** Останавливает эмулятор и переходит в режим редактирования.

7.1.1.6 Сервис

Элемент главного меню **Сервис** содержит пункты:

- **Проверить изображение.** Проверяет по текущей базе данных, имеются ли ошибки в описании объектов, включенных в текущее изображение;
- **Проверить все.** Проверяет по всем изображениям текущей базы данных, имеются ли ошибки в описании объектов;
- **Редактор рабочих столов.** Открывает редактор рабочих столов операторской станции.

7.1.1.7 Окна

Элемент главного меню **Окна** содержит пункты:

- **Каскад.** Располагает открытые окна изображений в виде каскада;
- **Мозаика горизонтальная.** Располагает открытые окна изображений одно под другим, выравнивая их по ширине экрана;
- **Мозаика вертикальная.** Располагает открытые окна изображений слева направо, выравнивая их по высоте экрана;
- **Упорядочить значки.** Упорядочивает минимизированные окна изображений;

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

- **Упорядочить плавающие окна.** В режиме эмулятора упорядочивает все открытые плавающие окна;
- **Заккрыть все.** Закрывает все открытые окна;
- **Заккрыть плавающие окна.** В режиме эмулятора закрывает все открытые плавающие окна;
- **Список открытых изображений.** В меню добавлены все открытые изображения. Активное изображение отмечено флагом. Выбрав соответствующее изображение можно сделать его активным.

7.1.2 Панель инструментов для редактирования изображения. Панель инструментов для редактирования изображения состоит из набора кнопок (рисунок 31), которые в основном дублируют функции главного меню и обеспечивают более быстрый доступ к ним.



Рисунок 31 – Панель инструментов для редактирования изображения

Панель инструментов содержит следующие кнопки:

- 1 Открыть. Обеспечивает быстрый доступ к пункту Открыть меню Файл. Выбрать изображение требуемого типа, можно нажав на одну из появившихся кнопок (рисунок 32).

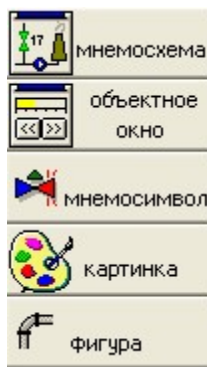








Рисунок 32 – Меню Файл / Открыть

- 2 Сохранить. Обеспечивает быстрый доступ к пункту Сохранить меню Файл.
- 2 Вырезать. Обеспечивает быстрый доступ к пункту Вырезать меню Правка.
- 3 Копировать. Обеспечивает быстрый доступ к пункту Копировать меню Правка.
- 4 Вставить. Обеспечивает быстрый доступ к пункту Вставить меню Правка.
- 5 Отменить. Обеспечивает быстрый доступ к пункту Отменить меню Правка.
- 6 Восстановить. Обеспечивает быстрый доступ к пункту Восстановить меню Правка.
- 7 Запрет перемещения. Запрещает(при первом нажатии) или разрешает(при повторном нажатии) перемещать элементы по поверхности изображения.
- 8 Войти. Позволяет войти внутрь объединения или соединения элементов для редактирования отдельных элементов.
- 9 Выйти. Позволяет вернуться к редактированию изображения в целом.
- 10 Атрибуты. Обеспечивает быстрый доступ к пункту Атрибуты меню Преобразования.
- 11 Вперед. Обеспечивает быстрый доступ к пункту Вперед меню Преобразования.
- 12 Назад. Обеспечивает быстрый доступ к пункту Назад меню Преобразования.
- 13 Объединить. Обеспечивает быстрый доступ к пункту Объединить меню Преобразования.
- 14 Разъединить. Обеспечивает быстрый доступ к пункту Разъединить меню Преобразования.
- 15 Поворот. Обеспечивает быстрый доступ к пункту Поворот меню Преобразования.

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

- 16  По горизонтали. Обеспечивает быстрый доступ к пункту Отражение – по горизонтали меню Преобразования.
- 17  По вертикали. Обеспечивает быстрый доступ к пункту Отражение – по вертикали меню Преобразования.
- 18  Старт. Обеспечивает быстрый доступ к пункту Старт меню Динамика.
- 19  Стоп. Обеспечивает быстрый доступ к пункту Стоп меню Динамика.
- 20  Проверить изображение. Обеспечивает быстрый доступ к пункту Проверить изображение меню Сервис.
- 21  Проверить все. Обеспечивает быстрый доступ к пункту Проверить все меню Сервис.
- 22  Примитивы. Обеспечивает быстрый доступ к пункту Примитивы меню Сервис.
- 23  Аниматоры. Обеспечивает быстрый доступ к пункту Аниматоры меню Сервис.
- 24  Рецепторы. Обеспечивает быстрый доступ к пункту Рецепторы меню Сервис.
- 25  Вид мнемосхемы. Обеспечивает быстрый доступ к пункту Вид мнемосхемы меню Сервис.
- 26  Программа. Открывает окно редактирования встроенной программы для выбранного элемента.

















7.1.3 Панель инструментов для редактирования элементов изображения

Панель инструментов для редактирования элементов изображения состоит из набора кнопок (рисунок 33).



Рисунок 33 – Панель инструментов для редактирования элементов изображения

Панель содержит следующие кнопки:

- 1  **Выбор элемента.** Включает режим выделения элементов. Пока не выбран другой инструмент, Графит находится именно в этом режиме.
- 27  **Вращение.** Включает режим поворота элементов на произвольный угол.
- 28  **Форма.** Включает режим изменения формы элементов.
- 29  Прокрутка. Выключает режим прокрутки, если у окна изображения есть линии прокрутки. Выберите режим, нажмите левую кнопку мыши в поле изображения и не отпуская ее потяните, так чтобы стали видны недоступные части изображения.
- 30  **Отрезок.** Включает режим рисования отрезка.
- 31  **Прямоугольник.** Включает режим рисования прямоугольник.
- 32  **Ломаная.** Включает режим рисования ломаной линии с произвольными углами наклона линий.
- 33  **Ломаная с горизонтальными ступенями.** Включает режим рисования ломаной линии с горизонтальными ступенями
- 34  **Ломаная с вертикальными ступенями.** Включает режим рисования ломаной линии с вертикальными ступенями.
- 35  **Кривая.** Включает режим рисования кривых линий.
- 36  **Эллипс.** Включает режим рисования круга.
- 37  **Сектор.** Включает режим рисования сектора круга.
- 38  **Сегмент.** Включает режим рисования сегмента круга.
- 39  **Дуга.** Включает режим рисования дуги окружности.
- 40  **Текст.** Включает режим добавления текста в изображение.
- 41  **Реквизит.** Включает режим добавления реквизита в изображение.

Инд. № подл.	Подп. и дата	Инд. №	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата	Подп. и дата	
Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5			Лист 96

- 42  **Цифры.** Включает режим добавления цифровой информации в изображение. Элемент Цифры представляет собой элемент Текст, для которого определены аниматоры. В оперативном режиме элемент отображает изменение числовых значений соответствующих параметров.
- 43  **Дата-время.** Включает режим добавления в изображение элемента для отображения даты и времени.
- 44  **Строка состояния.** Включает режим добавления строки состояния в изображение.
- 45  **Отросток.** Включает режим добавления отростка в изображение.
- 46  **Кнопка.** Преобразует в кнопку выбранные элементы изображения.
- 47  **Список состояний.** Включает режим добавления списка состояний объекта в изображение.
- 48  **Список изображений.** Включает режим добавления списка изображений.
- 49  **Вставка.** Включает режим вставки заранее подготовленных изображений. Выбрать тип вставляемого изображения, можно нажав одну из кнопок, изображенных на рисунке 32.
- 50  **Серия.** Включает режим создания серии элементов, на основе выбранных.
- 51  **Уровень.** Включает режим создания уровня на основе выбранных элементов.
- 52  **Движок.** Включает режим создания движка на основе выбранных элементов.
- 53  **Ползунок.** Включает режим создания ползунка на основе выбранных элементов.
- 54  **Соединение.** Осуществляет операцию соединения выбранных элементов.
- 55  **Столбик.** Включает режим добавления уровня прямоугольной формы.
- 56  **Шкала.** Включает режим добавления линейной шкалы.
- 57  **Круговая шкала со стрелкой.** Включает режим добавления круговой шкалы -  или комбинации(объединения) круговой шкалы со стрелкой.
- 58  **Дополнительные элементы.** Позволяет выбрать из списка и установить в изображение один из следующих элементов:

- График;
- Таблица данных из архива;
- Состояния архивных станций;
- Список параметров;
- Окно ошибок;
- Полис;
- Хроника дискретных событий;
- ActiveX.

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата

7.2 Справочник по встроенной программе

7.2.1 Константы

Значения констант приведены в таблице 29.

Таблица 29 – Значения констант

Выражение	Значение	Примечание
True	1	Истина
False	0	Ложь
Pi	3,1415...	Число «Пи»
E	2,7182...	Число «Е»

7.2.2 Функции

7.2.2.1 Математические

Основные математические функции приведены в таблице 30. Они импортированы из **RunTime** библиотеки **Borland C**.

Таблица 30 – Математические функции

Выражение	Название
Abs	Абсолютная величина числа
ACos	Арккосинус
ASin	Арсинус
Atan	Арктангенс
Ceil	Округление числа до ближайшего целого сверху
Cos	Косинус
Cosh	Гиперболический косинус
Exp	Экспонента
Floor	Целая часть числа
Log	Натуральный логарифм
Log10	Десятичный логарифм
Max	Наибольшее из двух чисел
Min	Наименьшее из двух чисел
Mod	Остаток от деления одного числа на другое
Pow	Возведение в степень
Rand	Генератор случайных чисел
Sin	Синус
Sinh	Гиперболический синус
Sqrt	Извлечение квадратного корня
Tan	Тангенс
Tanh	Гиперболический тангенс

7.2.2.1.1 Abs

Абсолютная величина числа

```
function Abs(X: double): double;
```

Параметры: X. Исходное число

Возвращает: X, если $X \geq 0$, -X, если $X < 0$

Смотри также: Min, Max

Инд. №	Подп. и дата
Взаим. инв. №	Подп. и дата
Инд. № подл.	Изм Лист № докум Подп. Дата

7.2.2.1.2 ACos

Арккосинус

```
function ACos(X: double): double;
```

Параметры: X. Исходное число в диапазоне от -1 до 1

Возвращает: Число Y в диапазоне от 0 до pi, такое что $\cos(Y)=X$.

Смотри также: Cos, ASin.

7.2.2.1.3 ASin

Арксинус

```
function ASin(X: double): double;
```

Параметры: X. Исходное число в диапазоне от -1 до 1

Возвращает: Число Y в диапазоне от $-\pi/2$ до $\pi/2$, такое что $\sin(Y)=X$.

Смотри также: ACos, Sin.

7.2.2.1.4 Atan

Арктангенс

```
function Atan(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Арктангенс X, т.е. такое число Y, что $\text{tg}(Y) = X$. Возвращаемое значение лежит в пределах от $-\pi/2$ до $\pi/2$.

7.2.2.1.5 Ceil

Округление числа до ближайшего целого сверху

```
function Ceil(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Наименьшее целое Y, такое что $Y \geq X$.

См. также: Floor.

7.2.2.1.6 Cos

Косинус

```
function Cos(X: double): double;
```

Параметры: X. Исходное число в радианах

Возвращает: Косинус угла X

Смотри также: Sin, ACos, Cosh.

7.2.2.1.7 Cosh

Гиперболический косинус

```
function Cosh(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Гиперболический косинус X, т.е. $(e^X + e^{-X}) / 2$.

Смотри также: Cos, Sinh.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.2.1.8 Exp

Экспонента

```
function Exp(X: double): double;
```

Параметры: X. Исходное число

Возвращает: e^X .

Смотри также: Log, Pow.

7.2.2.1.9 Floor

Целая часть числа

```
function Floor(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Наибольшее целое Y, такое что $Y \leq X$.

Смотри также: Ceil

7.2.2.1.10 Log

Натуральный логарифм

```
function Log(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Натуральный логарифм X, т.е. такое число Y, что $e^Y = X$.

Смотри также: Log10, Exp, Pow.

7.2.2.1.11 Log10

Десятичный логарифм

```
function Log10(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Десятичный логарифм X, т.е. такое число Y, что $10^Y = X$.

Смотри также: Log, Pow.

7.2.2.1.12 Max

Наибольшее из двух чисел

```
function Max(X: double; Y: double): double;
```

Параметры: X. Первое число, Y. Второе число

Возвращает: Максимум из X и Y

Смотри также: Min, Abs.

7.2.2.1.13 Min

Наименьшее из двух чисел

```
function Min(X: double; Y: double): double;
```

Параметры: X. Первое число, Y. Второе число

Возвращает: Минимум из X и Y

Смотри также: Max, Abs.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.2.1.14 Mod

Остаток от деления одного числа на другое

```
function Mod(X: double; Y: double): double;
```

Параметры: X. Делимое число, Y. Делитель

Возвращает: Остаток от деления X на Y, т.е. такое число Z, что $X = N \cdot Y + Z$, где N - целое и $0 \leq Z < Y$.

Смотри также: Floor.

7.2.2.1.15 Pow

Возведение в степень

```
function Pow(X: double; Y: double): double;
```

Параметры: X. Число, возводимое в степень, Y. Степень

Возвращает: X^Y

Смотри также: Exp, Log

7.2.2.1.16 Rand

Генератор случайных чисел

```
function Rand: double;
```

Возвращает: Случайное число, равномерно распределенное от 0 до 1

7.2.2.1.17 Sin

Синус

```
function Sin(X: double): double;
```

Параметры: X. Исходное число в радианах

Возвращает: Синус угла X

Смотри также: ASin, Cos, Sinh

7.2.2.1.18 Sinh

Гиперболический синус

```
function Sinh(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Гиперболический синус X, т.е. $(e^X - e^{-X}) / 2$.

Смотри также: Cosh, Sin.

7.2.2.1.19 Sqrt

Извлечение квадратного корня

```
function Sqrt(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Квадратный корень из X

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.2.1.20 Tan

Тангенс

```
function Tan(X: double): double;
```

Параметры: X. Исходное число в радианах

Возвращает: Тангенс угла X

Смотри также: Tanh, Atan, Cos, Sin.

7.2.2.1.21 Tanh

Гиперболический тангенс

```
function Tanh(X: double): double;
```

Параметры: X. Исходное число

Возвращает: Гиперболический тангенс X, т.е. $\sinh(X) / \cosh(X)$.

Смотри также: Tan, Sinh, Cosh.

7.2.2.2 Работа со строками

Функции преобразования строк приведены в таблице 31.

Таблица 31 - Функции преобразования

Выражение	Примечание
CompareText	Сравнивает 2 строки не различая строчные и прописные буквы
Copy	Выделяет подстроку из заданной строки
Pos	Ищет вхождение подстроки в заданную строку
String	Преобразует число в строку
StrToFloat	Преобразует строку в число с плавающей точкой
Строка	Преобразует число в строку

7.2.2.2.1 CompareText

Сравнивает 2 строки не различая строчные и прописные буквы

```
function CompareText(S1: string; S2: string): double;
```

Параметры: S1. Первая строка, S2. Вторая строка

Возвращает: -1, если s1 идет раньше s2 по алфавиту; 1, если s1 идет позже s2 по алфавиту; 0, если s1 совпадает с s2 .

7.2.2.2.2 Copy

Выделяет подстроку из заданной строки

```
function Copy(Source: string; Position: double; Len: double): string;
```

Параметры: Source. Исходная строка, Position. Номер символа (начиная с 1) с которого должна начинаться подстрока, Len. Длина выделяемой подстроки

Возвращает: Выделенную подстроку.

Смотри также: Pos.

7.2.2.2.3 Pos

Ищет вхождение подстроки в заданную строку

```
function Pos(Substr: string; S: string): double;
```

Параметры: Substr. Искомая подстрока, S. Исходная строка.

Возвращает: Позицию, с которой начинается найденная подстрока (начиная с 1) .
Если подстрока не найдена, возвращает 0

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

Смотри также: Copy

7.2.2.2.4 String

Преобразует число в строку

```
function String(X: double): string;
```

Параметры: X. Исходное число

Возвращает: Представление числа с плавающей точкой в десятичном виде.

Смотри также: StrToFloat , TimeToString , Строка

7.2.2.2.5 StrToFloat

Преобразует строку в число с плавающей точкой

```
function StrToFloat(X: string): double;
```

Параметры: X. Исходная строка в формате [пробелы] [знак] [ddd] [.] [ddd] [e]E[знак]ddd], где d обозначает цифру.

Возвращает: Полученное число

Смотри также: String.

7.2.2.2.6 Строка

Преобразует число в строку

```
function Строка(X: double; N: double): string;
```

Параметры: X. Исходное число, N. Требуемое количество знаков после десятичной точки.

Возвращает: Представление числа X в десятичной форме с N знаками после десятичной точки.

Смотри также: String, TimeToString , StrToFloat.

7.2.2.3 Работа с временем

Позволяют работать с астрономическим временем. Функции времени приведены в таблице 32.

Таблица 32 – Функции времени

Выражение	Примечание
LocalTime	Местное время
Time	Время по Гринвичу
TimeToString	Преобразует время в строку
SetTimer	Взводит один из таймеров

7.2.2.3.1 LocalTime

Местное время

```
function LocalTime: double;
```

Возвращает: Время в секундах с 01.01.1970

Смотри также: Time , TimeToString.

7.2.2.3.2 Time

Время по Гринвичу

```
function Time: double;
```

Возвращает: Время в секундах с 01.01.1970

Смотри также: LocalTime, TimeToString.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.2.3.3 TimeToString

Преобразует время в строку

```
function TimeToString(X: double; ShowDate: double; ShowTime: double; ShowMs: double): string;
```

Параметры:

- **X**. Исходное время в секундах с 01.01.1970
- **ShowDate**. Если ShowDate <> 0 то в строковое представление включается дата в формате, определенном в настройках Windows
- **ShowTime**. Если ShowTime <> 0 то в строковое представление включается время суток в формате, определенном в настройках Windows
- **ShowMs**. Если ShowMs <> 0 то в строковое представление включаются миллисекунды, содержащиеся в дробной части числа X.

Возвращает: Строковое представление времени

Смотри также: Time, LocalTime.

7.2.2.3.4 SetTimer

Взводит один из таймеров

```
procedure SetTimer(TimerID: double; TimeInSeconds: double);
```

Параметры:

- **TimerID**. Номер взводимого таймера. Таймеры нумеруются целыми числами 1, 2, 3...
- **TimeInSeconds**. Время, через которое таймер должен сработать. Время задается в секундах, допускаются дробные значения. Если передать значение 0, то данный таймер будет деактивирован.

Примечание - Когда таймер срабатывает, вызывается обработчик события **OnTimer**. Номер сработавшего таймера передается в процедуру OnTimer в качестве параметра.

Таймер срабатывает однократно. Если требуется периодическое выполнение некоторого действия, следует в обработчике **OnTimer** повторно вызвать **SetTimer**.

Повторный вызов **SetTimer(N, T)** с тем же номером N переустанавливает N-й таймер (т.е. он сработает через T секунд после второго вызова SetTimer). Взведенный таймер можно отменить, вызвав SetTimer(N, 0).

В отличие от **OnDataChange**, обработчик **OnTimer** будет вызываться, даже если никакие значения параметров на мнемосхеме не изменяются.

Смотри также: OnTimer.

7.2.2.4 Работа со звуком

Позволяют управлять очередью звуковых файлов. Функции управления приведены в таблице 33.

Таблица 33 – Функции управления звуками

Выражение	Примечание
Звук	Устаревшая. Используйте функции ЗвукВыключить, ЗвукВключить и т.д.
ЗвукВключен	Позволяет определить, проигрывается ли звук в настоящий момент.
ЗвукВключить	Воспроизводит звук из WAV-файла через звуковую карту.
ЗвукВыключить	Выключает конкретный звук.
ЗвукВыключитьВсе	Выключает все звуки с указанным приоритетом

Инд. №	Подп. и дата
Взаим. инв. №	
Инд. № подл.	

7.2.2.4.1 Звук

Устаревшая. Используйте функции **ЗвукВключить**, **ЗвукВыключить** и т.д.

```
procedure Звук(FileName: string; Loop: double; Priority: double);
```

Параметры:

- **FileName.** Это имя WAV-файла, содержащего требуемый звук. Если указывается пустая строка, то текущий звук выключается. Имя файла указывается с расширением, но без пути: файл берется из каталога звуков, указанного в настройках. Если каталог звуков не указан, то он совпадает с рабочим каталогом.
- **Loop.** Если равен 0, то звук проигрывается однократно, если равен 1, то звук зацикливается
- **Priority.** Зарезервирован для будущего использования. Следует устанавливать равным 0.

Смотри также: **ЗвукВключить**, **ЗвукВыключить**, **ЗвукВыключитьВсе**, **ЗвукВключен**.

7.2.2.4.2 ЗвукВключен

Позволяет определить, проигрывается ли звук в настоящий момент.

```
function ЗвукВключен(SoundId: double): double;
```

Параметры: **SoundId.** Идентификатор звука, полученный с помощью функции **ВключитьЗвук**, или равен 0.

Возвращает:

- **0.** Звука в очереди нет;
- **1.** Звук есть в очереди, но не проигрывается в настоящий момент;
- **2.** Звук проигрывается в настоящий момент.

Если **SoundId** не равен 0, возвращаемое значение относится к этому звуку; если **SoundId = 0** - то ко всем звукам

Смотри также: **ЗвукВключить**, **ЗвукВыключить**, **ЗвукВыключитьВсе**.

7.2.2.4.3 ЗвукВключить

Воспроизводит звук из WAV-файла через звуковую карту.

```
function ЗвукВключить(FileName: string; Loop: double; Priority: double; Period: double): double;
```

Параметры:

- **FileName.** Имя WAV-файла, содержащего требуемый звук. Если указывается пустая строка, то текущий звук выключается. Имя файла указывается с расширением, но без пути: файл берется из каталога звуков, указанного в настройках. Если каталог звуков не указан, то он совпадает с рабочим каталогом.
- **Loop.** Если **Loop** равен 0, звук воспроизводится однократно, если **Loop** равен 1, звук зацикливается.
- **Priority.** Приоритет звука (число от 1 до 10). Звуки с более высоким приоритетом вытесняют звуки с более низким приоритетом.
- **Period.** Период повтора зацикленного звука в миллисекундах. Имеет смысл, если **Loop** равен 1. В этом случае звук будет повторяться с периодом, равным $\max(\text{Period}, \text{Length})$, где **Length** - продолжительность самого звука. В частности, при **Period** равным 0 звук будет повторяться без пауз.

Возвращает: Идентификатор звука.

Смотри также: **ЗвукВыключить**, **ЗвукВыключитьВсе**, **ЗвукВключен**.

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

7.2.2.4.4 ЗвукВыключить

Выключает конкретный звук.

```
function ЗвукВыключить(SoundId: double): double;
```

Параметры: *SoundId*. Идентификатор звука, полученный с помощью функции ВключитьЗвук.

Возвращает: 1, если звук с указанным идентификатором присутствует в очереди, иначе 0.

Смотри также: ЗвукВключить, ЗвукВыключитьВсе, ЗвукВключен.

7.2.2.4.5 ЗвукВыключитьВсе

Выключает все звуки с указанным приоритетом

```
function ЗвукВыключитьВсе(Priority: double; StopLower: double): double;
```

Параметры:

- **Priority**. Приоритет выключаемых звуков (число от 1 до 10).
- **StopLower**. Если StopLower = 0, выключаются все звуки с приоритетом Priority, если StopLower = 1, выключаются все звуки с приоритетом Priority и ниже.

Возвращает: 1, если хотя бы один звук был выключен, иначе 0.

Смотри также: ЗвукВключить, ЗвукВыключить, ЗвукВключен.

7.2.2.5 Работа с INI-файлами

Функции, приведенные в таблице 34, позволяют читать и записывать значения вспомогательных параметров в INI-файлы.

Таблица 34 - Функции работы с INI-файлами

Выражение	Примечание
DeleteIniSection	Удаляет секцию из INI-файла вместе с ее содержанием
DeleteIniString	Удаляет параметр из INI-файла
ReadIniString	Считывает значение параметра из INI-файла
WriteIniString	Записывает значение параметра в INI-файл.

7.2.2.5.1 DeleteIniSection

Удаляет секцию из INI-файла вместе с ее содержанием

```
function DeleteIniSection(FileName: string; Section: string): double;
```

Параметры:

- **FileName**. Полное имя INI-файла. Если имя указывается без пути, то файл размещается в каталоге Windows.
- **Section**. Имя секции INI-файла.

Возвращает: 1 - в случае успеха, 0 - в случае ошибки.

Смотри также: DeleteIniString.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.2.5.2 DeleteIniString

Удаляет параметр из INI-файла

```
function DeleteIniString(FileName: string; Section: string; Param: string): double;
```

Параметры:

- **FileName.** Полное имя INI-файла. Если имя указывается без пути, то файл размещается в каталоге Windows.
- **Section.** Имя секции INI-файла.
- **Param.** Имя параметра

Возвращает: 1 - в случае успеха, 0 - в случае ошибки.

Смотри также: DeleteIniSection, WriteIniString.

7.2.2.5.3 ReadIniString

Считывает значение параметра из INI-файла

```
function ReadIniString(FileName: string; Section: string; Param: string; DefaultValue: string): string;
```

Параметры:

- **FileName.** Полное имя INI-файла. Если имя указывается без пути, то файл размещается в каталоге Windows.
- **Section.** Имя секции INI-файла.
- **Param.** Имя параметра
- **DefaultValue.** Значение по умолчанию (возвращается, если не найдены соответствующие файл, секция или параметр).

Возвращает: Значение, содержащееся в указанной строке справа от знака =.

Смотри также: WriteIniString, StrToFloat.

7.2.2.5.4 WriteIniString

Записывает значение параметра в INI-файл.

```
function WriteIniString(FileName: string; Section: string; Param: string; Value: string): double;
```

Параметры:

- **FileName.** Полное имя INI-файла. Если имя указывается без пути, то файл размещается в каталоге Windows.
- **Section.** Имя секции INI-файла.
- **Param.** Имя параметра
- **Value.** Значение параметра

Возвращает: 1 - в случае успеха, 0 - в случае ошибки.

Смотри также: ReadIniString, DeleteIniString, DeleteIniSection, TimeToString.

7.2.2.6 Прочие функции

Прочие функции приведены в таблице 35.

Таблица 35 - Прочие функции

Выражение	Примечание
ИмяПользователя	Получить имя текущего пользователя
ОткрытьОкно	Открывает окно на операторской станции.
ПользовательВГруппе	Позволяет определить, принадлежит ли текущий пользователь указанной группе.

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

7.2.2.6.4 ВвестиЧисло

Вызывает окно диалога редактирования числа пользователем.

```
function ВвестиЧисло (var Число: double; Мин, Макс, Точность: double; ЕДИзм,
Заголовок: string): double;
```

Параметры:

- **Число.** Редактируемое число. Значение передается по ссылке и изменяется при закрытии диалогового окна по кнопке **Ввод**;
- **Мин.** Минимальное допустимое значение;
- **Макс.** Максимальное допустимое значение;
- **Точность.** Количество десятичных цифр дробной части в строковом представлении числа. Введенное число округляется с указанной точностью. Если указать отрицательную *Точность*, младшие разряды целой части будут заменяться нулями;
- **ЕДИзм.** Единица измерения. Выводится в окне в виде текста;
- **Заголовок.** Заголовок окна диалога.

Возвращает: 1, если диалог был закрыт по кнопке Ввод, 0 – в противном случае.

7.2.2.7 Обработчики событий

Обработчик событий - это процедура с зарезервированным именем (**OnOpen, OnClose, OnDataChange, OnMouse, OnKey, OnTimer**). Такая процедура (если она есть) вызывается извне программы при возникновении определенного события, в то время как прочие процедуры могут быть вызваны только из самой программы. Процедуры обработчика событий сведены в таблицу 36.

Таблица 36 – Процедуры обработчика событий

Выражение	Примечание
OnClose	Вызывается при закрытии окна с изображением, к которому относится элемент с программой.
OnDataChange	Вызывается, если изменились значения каких-либо параметров объектов или системы (необязательно связанных с данным окном).
OnKey	Вызывается при нажатии или отпуске любых клавиш на клавиатуре. В настоящей версии не поддерживается.
OnMouse	Вызывается при нажатии, отпуске кнопок или перемещении мыши в зоне данного элемента.
OnOpen	Вызывается при открытии окна с изображением, к которому относится элемент с программой.
OnTimer	Вызывается при срабатывании таймера, установленного функцией SetTimer

7.2.2.7.1 OnClose

Вызывается при закрытии окна с изображением, к которому относится элемент с программой.

```
procedure OnClose;
```

Смотри также: OnOpen.

7.2.2.7.2 OnDataChange

Вызывается, если изменились значения каких-либо параметров объектов или системы (необязательно связанных с данным окном).

```
procedure OnDataChange;
```

Инв. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инв. № подл.	

7.2.2.7.3 OnKey

Вызывается при нажатии или отпускании любых клавиш на клавиатуре. В настоящей версии не поддерживается.

```
procedure OnKey(eventType, key, modKeys: double);
```

Параметры:

- **EventType.** Тип события. Допустимы следующие значения:
 - а) **evKeyDown.** Нажатие клавиши;
 - б) **evKeyUp.** Отпускание клавиши;
 - в) **evKeyAuto.** Вызывается периодически, пока пользователь держит клавишу нажатой;
 - г) **evKeyCancel.** Вызывается, когда элемент, принявший ранее событие нажатия клавиши, становится неактивным, или когда содержащее его окно теряет фокус;
- **Key.** Код клавиши. Узнать код конкретной клавиши можно в диалоге настроек события для рецептора;
- **ModKeys.** Состояние клавиш **Ctrl** и **Shift**

7.2.2.7.4 OnMouse

Вызывается при нажатии, отпускании кнопок или перемещении мыши в зоне данного элемента.

```
procedure OnMouse(EventType, X, Y, ModKeys: double);
```

Параметры:

- **EventType.** Тип события. При нажатии левой кнопки мыши в зоне элемента этот элемент начинает перехватывать все события мыши вплоть до ее отпускания, даже если они произошли вне зоны этого элемента. Возможные типы события:
 - а) **evMouseDown.** Нажатие левой кнопки мыши в зоне данного элемента;
 - б) **evMouseUp.** Отпускание левой кнопки мыши;
 - в) **evMouseMove.** Перемещение указателя мыши;
 - г) **evMouseEnter.** "Въезд" указателя мыши в зону данного элемента (неважно, нажата левая кнопка или нет);
 - д) **evMouseLeave.** "Выезд" указателя мыши из зоны данного элемента (неважно, нажата левая кнопка или нет);
 - е) **evMouseAuto.** Вызывается периодически, пока пользователь держит нажатой левую кнопку мыши в зоне данного элемента;
 - ж) **evMouseCancel.** Вызывается, когда элемент, захвативший мышь, становится неактивным. После этого элемент перестает перехватывать события мыши;
 - з) **evMouseDbiClik.** Двойной щелчок левой кнопкой мыши в зоне данного элемента;
 - и) **evMenuPopup.** Нажатие правой кнопки мыши в зоне данного элемента;
- **X.** Координата X указателя мыши
- **Y.** Координата Y указателя мыши
- **ModKeys.** Состояние клавиш **Ctrl** и **Shift**

7.2.2.7.5 OnOpen

Вызывается при открытии окна с изображением, к которому относится элемент с программой.

```
procedure OnOpen;
```

Смотри также:

7.2.2.7.6 OnTimer

Вызывается при срабатывании таймера, установленного функцией **SetTimer**.

```
procedure OnTimer(timerID: double);
```

Параметры: *timerID.* Номер сработавшего таймера.

Смотри также: **SetTimer**.

Инд. № подл.	Подп. и дата
Взаим. инв. №	Инд. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист
						110

7.2.3 Доступ к элементу изображения

7.2.3.1 Атрибуты

Атрибуты элемента изображения, к которому относится программа, можно использовать в выражениях непосредственно по имени. Изменяемые свойства можно также использовать в левой части оператора присваивания. Все координаты являются мировыми (подробнее о координатах см. в 2.6.7). Функции атрибутов сведены в таблицу 37.

Таблица 37 - Функции атрибутов

Выражение	Примечание
ВерхняяГраница	Y-координата верхней стороны описанного около элемента прямоугольника
Видимость	Видимость элемента (изменяемое)
Высота	Высота описанного около элемента прямоугольника
ЗапретУправления	Запрет управления (т.е. срабатывание рецепторов) для элемента (изменяемое)
ЛеваяГраница	X-координата левой стороны описанного около элемента прямоугольника
МиганиеЗаполнения	Частота мигания заполнения (изменяемое)
МиганиеКонтура	Частота мигания контура (изменяемое)
МиганиеТекста	Частота мигания текста (изменяемое)
МиганиеФона	Частота мигания фона (изменяемое)
НижняяГраница	Y-координата нижней стороны описанного около элемента прямоугольника
НомерКадра	Номер кадра для элементов типа Картинка (изменяемое)
ПериодСменыКадра	Период смены кадра для элементов типа Картинка (изменяемое)
ПраваяГраница	X-координата правой стороны описанного около элемента прямоугольника
Текст	Значение текста (изменяемое)
ЦветЗаполнения	Цвет заполнения элемента (изменяемое)
ЦветКонтура	Цвет контура элемента (изменяемое)
ЦветТекста	Цвет текста элемента (изменяемое)
ЦветФона	Цвет фона элемента (изменяемое)
Ширина	Ширина описанного около элемента прямоугольника
ШкалЗнач	Уровень шкалы

7.2.3.1.1 ВерхняяГраница

Y-координата верхней стороны описанного около элемента прямоугольника

```
property ВерхняяГраница: double;
```

Смотри также: SetPosition.

7.2.3.1.2 Видимость

Видимость элемента (изменяемое)

```
property Видимость: double;
```

Возвращает: 0 - элемент видим, 1 - невидим

7.2.3.1.3 Высота

Высота описанного около элемента прямоугольника

```
property Высота: double;
```

7.2.3.1.4 ЗапретУправления

Запрет управления (т.е. срабатывание рецепторов) для элемента (изменяемое)

```
property ЗапретУправления: double;
```

Возвращает: 1 - установлен (рецепторы не работают), 0 - не установлен (рецепторы работают)

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инд. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

Изм	Лист	№ докум	Подп.	Дата	ПФДИ.421457.009 И3.5	Лист 111
-----	------	---------	-------	------	----------------------	-------------

7.2.3.1.5 ЛеваяГраница

X-координата левой стороны описанного около элемента прямоугольника

```
property ЛеваяГраница: double;
```

Смотри также: `SetPosition`.

7.2.3.1.6 МиганиеЗаполнения

Частота мигания заполнения (изменяемое)

```
property МиганиеЗаполнения: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.7 МиганиеКонтура

Частота мигания контура (изменяемое)

```
property МиганиеКонтура: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.8 МиганиеТекста

Частота мигания текста (изменяемое)

```
property МиганиеТекста: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.9 МиганиеФона

Частота мигания фона (изменяемое)

```
property МиганиеФона: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.10 НижняяГраница

Y-координата нижней стороны описанного около элемента прямоугольника

```
property НижняяГраница: double;
```

7.2.3.1.11 НомерКадра

Номер кадра для элементов типа Картинка (изменяемое)

```
property НомерКадра: double;
```

Возвращает: Номер кадра, начиная с 1.

7.2.3.1.12 ПериодСменыКадра

Период смены кадра для элементов типа Картинка (изменяемое)

```
property ПериодСменыКадра: double;
```

Возвращает: Значение периода в миллисекундах.

7.2.3.1.13 ПраваяГраница

X-координата правой стороны описанного около элемента прямоугольника

```
property ПраваяГраница: double;
```

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

7.2.3.1.14 Текст

Значение текста (изменяемое)

```
property Текст: string;
```

7.2.3.1.15 ЦветЗаполнения

Цвет заполнения элемента (изменяемое)

```
property ЦветЗаполнения: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.16 ЦветКонтура

Цвет контура элемента (изменяемое)

```
property ЦветКонтура: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.17 ЦветТекста

Цвет текста элемента (изменяемое)

```
property ЦветТекста: double;
```

Возвращает: (см. 7.2.3.3).

7.2.3.1.18 ЦветФона

Цвет фона элемента (изменяемое)

```
property ЦветФона: double;
```

Возвращает (см. 7.2.3.3).

7.2.3.1.19 Ширина

Ширина описанного около элемента прямоугольника

```
property Ширина: double;
```

7.2.3.1.20 ШкалЗнач

Атрибут **Уровень шкалы** определен для элементов типа **Уровень**, **Движок** и **Стрелка**. Эти элементы в разной форме отображают значение некоторой аналоговой величины. Ее значение должно меняться в пределах [0, 1].

```
property ШкалЗнач: double;
```

7.2.3.2 Доступ к элементам изображения

Элемент изображения (примитив), к которому относится программа, можно вызывать также как обычные процедуры. Методы элемента изображения сведены в таблицу 38.

Таблица 38 - Функции методов элемента изображения

Выражение	Примечание
GetPointX	Получить координаты маркера изменения формы
GetPointY	Получить координаты маркера изменения формы
SetPoint	Изменяет форму элемента изображения, устанавливая координаты маркера изменения формы
SetPosition	Перемещает элемента изображения, к которому относится программа.
ОтобразитьЧисло	УСТАРЕВШАЯ.

Инв. № подл.	Подп. и дата
	Инв. №
	Взаим. инв. №
	Подп. и дата

7.2.3.2.1 GetPointX

Получить координаты маркера изменения формы

```
function GetPointX(N: double): double;
```

Параметры: *N*. Номер маркера (нумерация начинается с 0).

Возвращает: Горизонтальную координату *N*-го маркера изменения формы для элемента изображения, к которому относится программа.

Смотри также: GetPointY, SetPoint.

7.2.3.2.2 GetPointY

Получить координаты маркера изменения формы

```
function GetPointY(N: double): double;
```

Параметры: *N*. Номер маркера (нумерация начинается с 0).

Возвращает: Вертикальную координату *N*-го маркера изменения формы для элемента изображения, к которому относится программа.

Смотри также: GetPointX, SetPoint.

7.2.3.2.3 SetPoint

Изменяет форму элемента изображения, устанавливая координаты маркера изменения формы

```
function SetPoint(N: double; X: double; Y: double): double;
```

Параметры:

- *N*. Номер маркера (нумерация начинается с 0);
- *X*. Горизонтальная координата;
- *Y*. Вертикальная координата.

Возвращает: 1 в случае успеха и 0 - в случае ошибки.

Смотри также: GetPointX, GetPointY, SetPosition.

7.2.3.2.4 SetPosition

Перемещает элемента изображения, к которому относится программа.

```
function SetPosition(X: double; Y: double): double;
```

Параметры:

- *X*. Горизонтальная координата точки, в которую должен быть перемещен левый верхний угол элемента;
- *Y*. Вертикальная координата точки, в которую должен быть перемещен левый верхний угол элемента.

Возвращает: 1 в случае успеха и 0 - в случае нулевого перемещения.

Смотри также: SetPoint .

7.2.3.2.5 ОтобразитьЧисло (устаревшее)

```
procedure ОтобразитьЧисло(X: double; N: double);
```

Параметры:

- *X*. Исходное число;
- *N*. Требуемое количество знаков после десятичной точки.

Смотри также: Строка

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

7.2.3.3 Кодировка цвета и мигания

Графит использует таблицу из **N** цветов и **M** осцилляторов (см. 2.6.10).

Цвета нумеруются числами от 0 до **N** – 1, а частоты миганий – числами от 1 до **M**, при этом частота мигания 0 означает ровное свечение. Во встроенной программе можно использовать эти числовые значения, например: `ЦветЗаполнения := 2; ЧастотаМигания := 1` означает, что заполнение примет цвет 3-го по счету элемента палитры, а частота мигания будет определяться первым осциллятором.

В диалоговом окне настройки палитры для каждому цвету и осциллятору можно дать наименование. Во встроенной программе эти наименования можно использовать как константы, например: `ЦветЗаполнения := Синий;`

В окне редактирования программы, при перетаскивании элемента палитры в поле программы, в текст добавляется наименование цвета, если оно указано, или код, если наименование отсутствует.

7.2.4 Доступ к параметрам объектов

Обращение к параметру объекта с маркой **марка** записывается следующим образом:

`Объект ('марка') . имя параметра`

После имени параметра можно через точку добавить свойство параметра:

`Объект ('марка') . имя параметра . НормЗнач`

Если параметр управляемый, это выражение может использоваться в левой части оператора присваивания.

При обращении к параметру объекта из объектного окна или мнемосимвола префикс `Объект ('марка')` можно опустить.

7.2.4.1 Свойства параметров

Для дискретного параметра `НормЗнач = РеалЗнач = ШкалЗнач`. Для аналогового параметра свойства связаны следующими соотношениями:

$$\text{РеалЗнач} = \text{НормЗнач} * (\text{РеалМакс} - \text{РеалМин}) + \text{РеалМин}$$

$$\text{НормЗнач} = (\text{РеалЗнач} - \text{РеалМин}) / (\text{РеалМакс} - \text{РеалМин})$$

$$\text{ШкалЗнач} = (\text{РеалЗнач} - \text{ШкалМин}) / (\text{ШкалМакс} - \text{ШкалМин})$$

$$\text{РеалЗнач} = \text{ШкалЗнач} * (\text{ШкалМакс} - \text{ШкалМин}) + \text{ШкалМин}$$

Аналогичными соотношениями связаны предустановки.

Свойства **РеалЗнач**, **НормЗнач** и **ШкалЗнач** являются изменяемыми, т.е. могут стоять в левой части оператора присваивания. Свойства параметров сведены в таблице 39.

Таблица 39 - Свойства параметров

Выражение	Примечание
Архивируется	Является архивируемым
Качество	Значение качества параметра
НормЗнач	Нормализованное значение (для аналогового параметра изменяется от 0 до 1)
НормПредуст	Нормализованная предустановка
РеалЗнач	Значение в физических единицах измерения
РеалМакс	Физический максимум
РеалМин	Физический минимум
РеалПредуст	Предустановка в физических единицах измерения
Статус	Статус параметра (Норма, НетЗначения, Изменен, ИзмененИПослан, Ошибка)
Управляется	Является управляемым
ШкалЗнач	Значение в диапазоне шкалы
ШкалМакс	Максимум шкалы
ШкалМин	Минимум шкалы
ШкалОсн	Основание шкалы
ШкалПредуст	Предустановка в диапазоне шкалы

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

7.2.4.1.1 Архивируется

Является архивируемым

```
property Архивируется: double;
```

7.2.4.1.2 Качество

Значение качества параметра (см. 5.1.5).

```
property Архивируется: double;
```

7.2.4.1.3 НормЗнач

Нормализованное значение (для аналогового параметра изменяется от 0 до 1)

```
property НормЗнач: double;
```

7.2.4.1.4 НормПредуст

Нормализованная предустановка

```
property НормПредуст: double;
```

7.2.4.1.5 РеалЗнач

Значение в физических единицах измерения

```
property РеалЗнач: double;
```

7.2.4.1.6 РеалМакс

Физический максимум

```
property РеалМакс: double;
```

7.2.4.1.7 РеалМин

Физический минимум

```
property РеалМин;
```

7.2.4.1.8 РеалПредуст

Предустановка в физических единицах измерения

```
property РеалПредуст: double;
```

7.2.4.1.9 Статус

Статус параметра (Норма, НетЗначения, Изменен, ИзменениПослан, Ошибка)

```
property Статус: double;
```

7.2.4.1.10 Управляется

Является управляемым

```
property Управляется: double;
```

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата
--------------	--------------	---------------	--------	--------------

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

7.2.4.1.11 ШкалЗнач

Значение в диапазоне шкалы

```
property ШкалЗнач: double;
```

7.2.4.1.12 ШкалМакс

Максимум шкалы

```
property ШкалМакс: double;
```

7.2.4.1.13 ШкалМин

Минимум шкалы

```
property ШкалМин: double;
```

7.2.4.1.14 ШкалОсн

Основание шкалы

```
property ШкалОсн: double;
```

7.2.4.1.15 ШкалПредуст

Предустановка в диапазоне шкалы

<Таблица>.<Поле> , например, Band.BandName

```
property ШкалПредуст: double;
```

7.2.4.2 Дополнительные атрибуты объектов

Для каждого типа объекта в приложении **Администратор БД** можно определить набор дополнительных атрибутов. Значения этих атрибутов для каждого объекта вводятся в приложении **Аркада**.

Обращение к атрибуту объекта с маркой «марка» записывается следующим образом:

```
Объект ('марка').Атрибут ('имя атрибута')
```

Тип данного выражения (таблица 40) зависит от типа атрибута, заданного в **Администраторе БД**.

Таблица 40 - Типы дополнительных атрибутов

Тип атрибута	Тип выражения
Строка	String
Число	Double
Дата	Не поддерживается

При обращении к атрибуту объекта из объектного окна или мнемосимвола префикс `Объект ('марка')` можно опустить.

7.2.5 Доступные поля базы данных (БД) Квинта

Значение поля БД в программе для объектного окна или мнемосимвола можно получить следующим образом:

```
<Имя таблицы>.<Имя поля>
```

Доступные таблицы и их поля перечислены в таблицах 41...48.

Например: `Band.BandName` означает имя узла, к которому принадлежит объект.

В обращении к полю БД из мнемосхемы нужно добавить впереди указание марки объекта в виде:

```
Объект ('NA004B1').Band.BandName
```

Инд. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инд. № подл.	

7.2.5.1 APag (диапазоны аналоговых параметров)

Таблица 41 - Поля APag, доступные из встроенной программы

Имя поля	Тип	Назначение
CardId	Double	Идентификатор объекта
ParamNo	Double	Номер параметра
S0	Double	0% шкалы
S100	Double	100% шкалы
SBase	Double	База шкалы
ParamMin	Double	Минимум диапазона
ParamMax	Double	Максимум диапазона
SComma	Double	Положение десятичной точки
ParamUnit	String	Единица измерения
TimeFormat	Double	Формат времени (1-сек,2-мин,3-часы,4-комбинированный:час/мин/сек)
DModify	Double	Дата и время последней правки

7.2.5.2 Band (узлы)

Таблица 42 - Поля Band, доступные из встроенной программы

Имя поля	Тип	Назначение
BandId	Double	Идентификатор узла
ParentBandId	Double	Идентификатор родительского узла
BandNo	Double	Порядковый номер узла
BandName	String	Имя узла
ShearMask	Double	Принадлежность срезам
Owner	String	Имя пользователя, создавшего объект
DCreate	Double	Дата и время создания
DModify	Double	Дата и время последней правки

7.2.5.3 Card (объекты)

Таблица 43 - Поля Card, доступные из встроенной программы

Имя поля	Тип	Назначение
CardId	Double	Идентификатор объекта
BandId	Double	Идентификатор узла
ObjTypeId	Double	Идентификатор типа объекта
Marka	String	Марка
TecTypeId	Double	Технологический тип объекта
ObjName	String	Имя объекта
ObjSign	String	Обозначение объекта
ContId	Double	Идентификатор контроллера
ObjAdr	Double	Адрес объекта (алгоблок-канал)
CardNo	Double	Номер объекта (для быстрого поиска)
ImageId	Double	Идентификатор изображения
BitNo	Double	Номер бита в запросе
EventSection	Double	Секция события
EventChannel	Double	Канал события
Flags	Double	Дополнительные признаки. Для двигателей Flags = 1 означает необходимость подсчета моточасов
DispName	String	Диспетчерское наименование
LocalNo	Double	Для ObjTypeId=22 используется как номер логической программы
Owner	String	Имя пользователя, создавшего объект
DCreate	Double	Дата и время создания
DModify	Double	Дата и время последней правки
ShearMask	Double	Принадлежность срезам
BlockId	Double	Идентификатор алгоблока
ModAdr	Double	В Квинте это поле не используется
CmdNo	Double	Служебное поле
EntryType	Double	Служебное поле

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Продолжение таблицы 43

Имя поля	Тип	Назначение
EntryNo	Double	Служебное поле
Channel	Double	Номер канала
QueryAdr	Double	Адрес запроса. Если не указан, в запросе используется ObjAdr
ClassId	Double	Идентификатор составного типа

7.2.5.4 ObjType (типы объектов)

Таблица 44 - Поля ObjType, доступные из встроенной программы

Имя поля	Тип	Назначение
ObjTypeId	Double	Идентификатор типа объекта
ObjShortName	String	Обозначение типа
ObjTypeName	String	Наименование типа
NeedExtAdr	Double	Необходимость указывать номер бита в адресе
DataSize	Double	Размер объектной области (байт)

7.2.5.5 Remicont (Ремиконты)

Таблица 45 - Поля Remicont, доступные из встроенной программы

Имя поля	Тип	Назначение
StationId	Double	Идентификатор абонента сети
ModelId	Double	Модель - ссылка на таблицу RemModel
Имя поля	Тип	Назначение
Version	Double	Номер версии
Dubl	Double	Наличие дублирования
CycleTime	Double	Время цикла
Имя поля	Тип	Назначение
FailureLimit	Double	Предельный уровень отказа
SyncTime	Double	Наличие синхронизации времени
DModify	Double	Дата и время последней правки
DCompile	Double	Дата последней компиляции ТП
DLoad	Double	Дата последней успешной загрузки ТП
DRebuild	Double	Дата последней сборки мусора
DChange	Double	Дата последнего изменения в ТП

7.2.5.6 Station (абоненты сети)

Таблица 46 - Поля Station, доступные из встроенной программы

Имя поля	Тип	Назначение
StationId	Double	Идентификатор абонента сети
StationName	String	Пользовательское имя
ComputerName	String	Имя в сети Microsoft Network
Addr	Double	Четырехбайтный адрес
StationType	Double	Тип станции 1-контроллер, 2-шлюз, 3-элемент кластера, 10-компьютер
Disabled	Double	Признак отключения
ContType	Double	Тип контроллера
BandId	Double	Узел
Owner	String	Имя пользователя, создавшего объект
DCreate	Double	Дата и время создания
DModify	Double	Дата и время последней правки
ShearMask	Double	Принадлежность срезам
WorkingShear	Double	Срез, используемый в оперативном режиме
Soft	Double	Запускаемое ПО
ArchStationId	Double	Идентификатор архивной станции
TimeStationId	Double	Идентификатор станции единого времени
WorkDir	String	Каталог с DAT-файлом на этом компьютере
ParentId	Double	Ид. кластера (только для StationType = 3)
SeqNumber	Double	Порядковый номер в пределах одного ParentId

Инв. №	Подп. и дата
Взаим. инв. №	
Подп. и дата	
Инв. № подл.	

7.2.5.7 ТесТуре (технологические подтипы)

Таблица 45 - Поля ТесТуре, доступные из встроенной программы

Имя поля	Тип	Назначение
ТесTypeId	Double	Технологич. тип объекта
ТесTypeName	String	Имя технологич. типа объекта
ObjTypeId	Double	Идентификатор оперативного типа объекта
Flag	Double	Доп. признаки
DModify	Double	Дата и время последней правки

7.2.5.8 ТРBlock (алгоблоки)

Таблица 48 - Поля ТРBlock, доступные из встроенной программы

Имя поля	Тип	Назначение
BlockId	Double	Ид. алгоблока (формируется автоматически)
TaskId	Double	Ид. задачи, к которой относится алгоблок
SeqNumber	Double	Порядковый номер алгоблока в задаче
ContId	Double	Ид. контроллера (это поле является избыточным, но нужно для поиска алгоблока по адресу)
Addr	Double	Адрес в контроллере
AlgNo	Double	Номер алгоритма для простого алгоблока или 0 - для сложного
ClassId	Double	Ид. класса, объектом которого является данный алгоблок или 0, если алгоблок - простой
Modifier	Double	Модификатор
BlockName	String	Имя алгоблока
DModify	Double	Дата последней модификации
X	Double	X-координата верхнего левого угла
Y	Double	Y-координата верхнего левого угла

Инд. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата

7.2.6 Синтаксические диаграммы

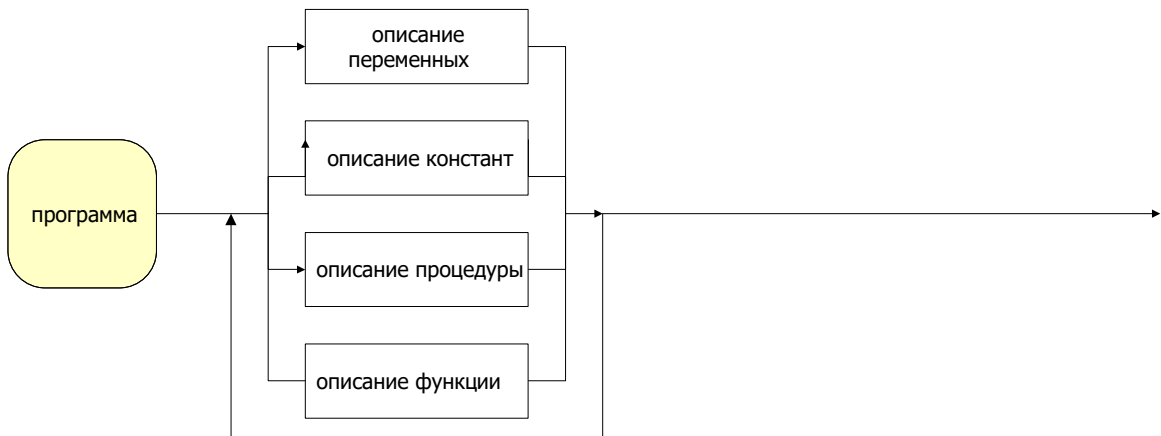


Рисунок 34 – Диаграмма описания программы

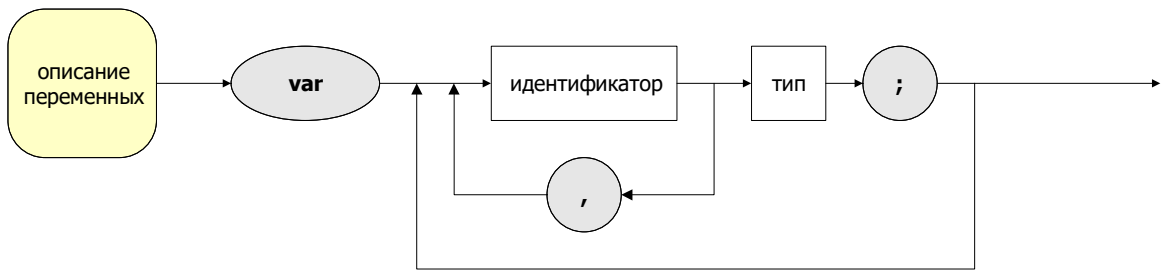


Рисунок 35 – Диаграмма описания переменных

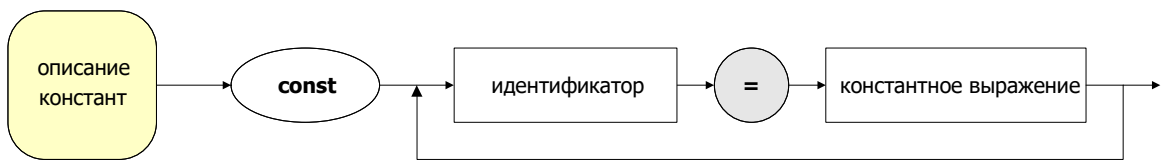


Рисунок 36 – Диаграмма описания констант

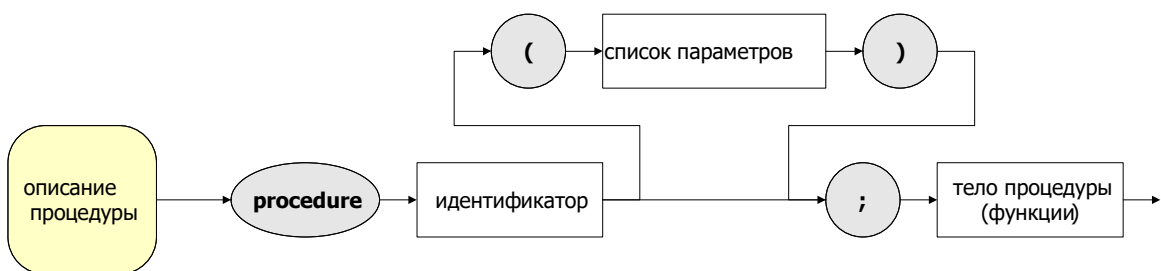


Рисунок 37 - Диаграмма описания процедуры

Инв. № подл.	Подп. и дата
Взаим. инв. №	Инв. №
Подп. и дата	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

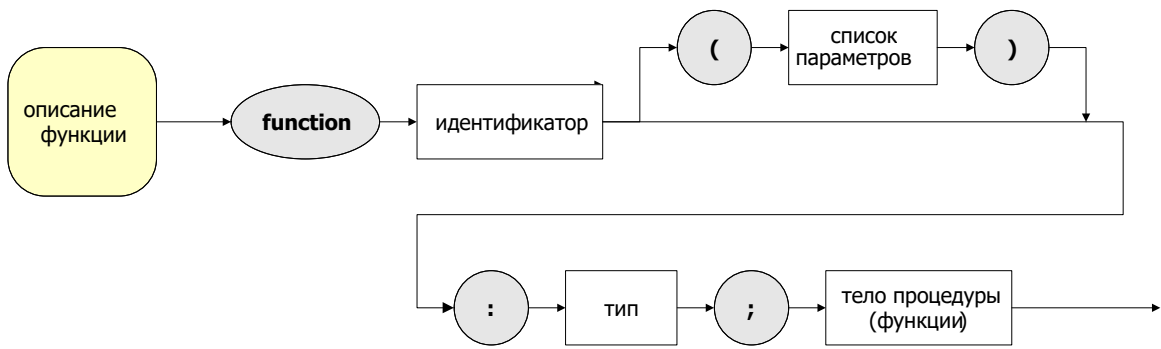


Рисунок 38 – Диаграмма описания функции

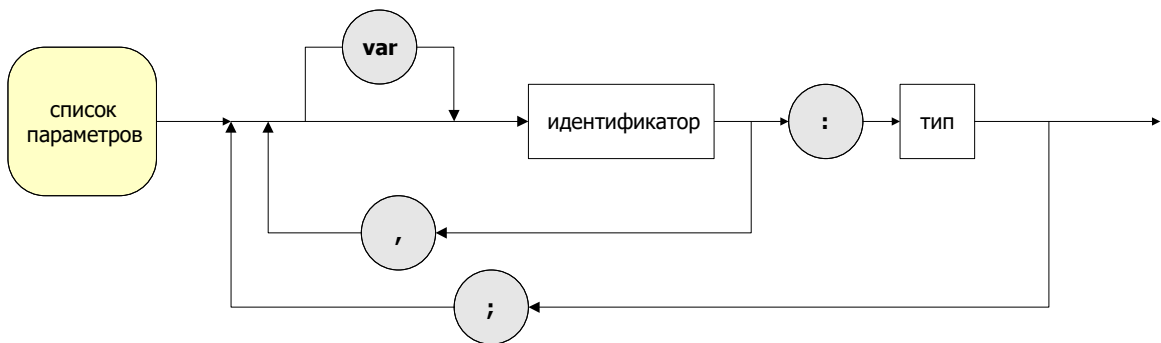


Рисунок 39 - Диаграмма списков параметров

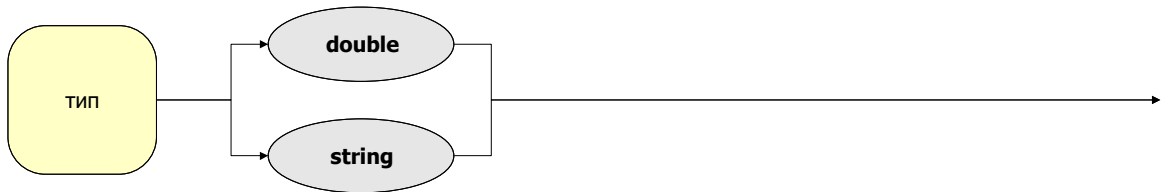


Рисунок 40 – Диаграмма описания типа



Рисунок 41 – Диаграмма описания утверждения

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата

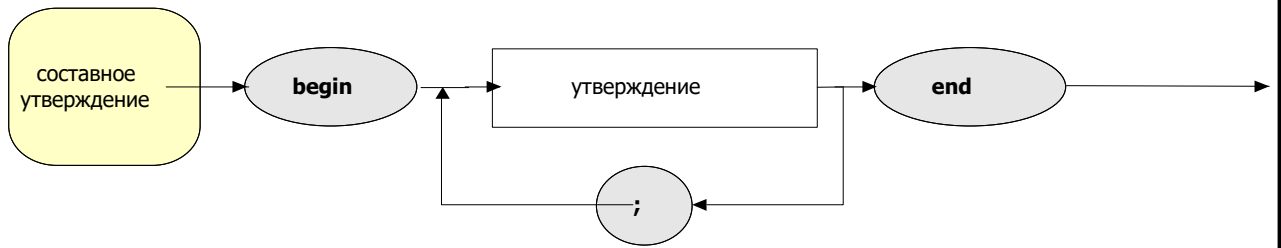


Рисунок 42 - Диаграмма описания составного утверждения

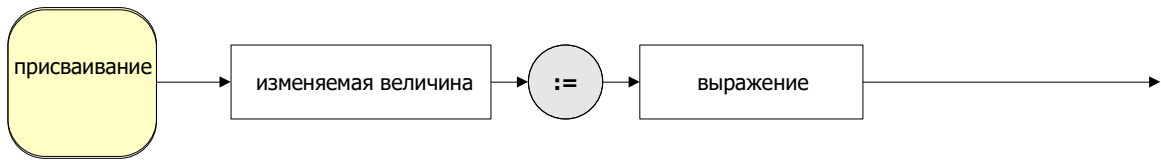


Рисунок 43 - Диаграмма описания присваивания

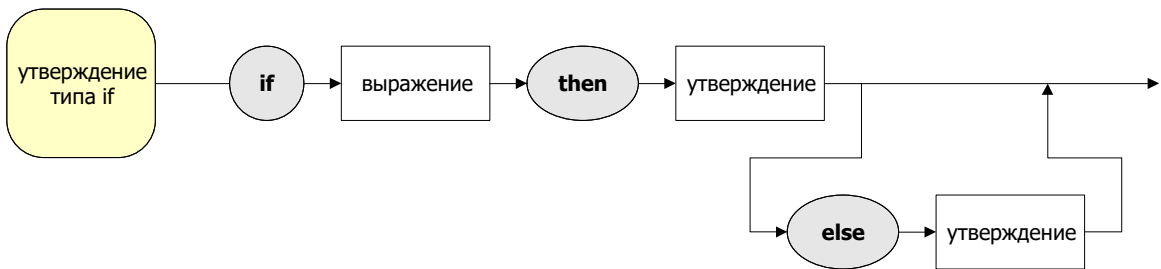


Рисунок 44 - Диаграмма описания утверждения типа IF

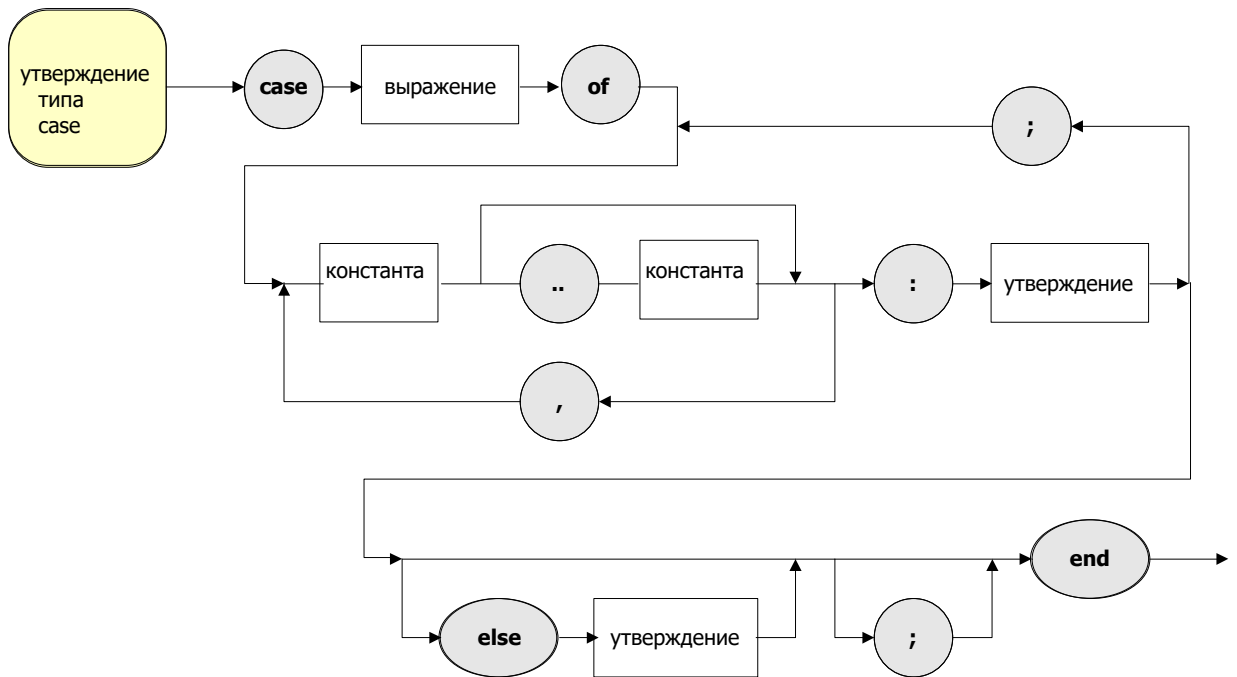


Рисунок 45 - Диаграмма описания утверждения типа CASE

Инв. № подл.	Подп. и дата	Подп. и дата
	Взаим. инв. №	Инв. №

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

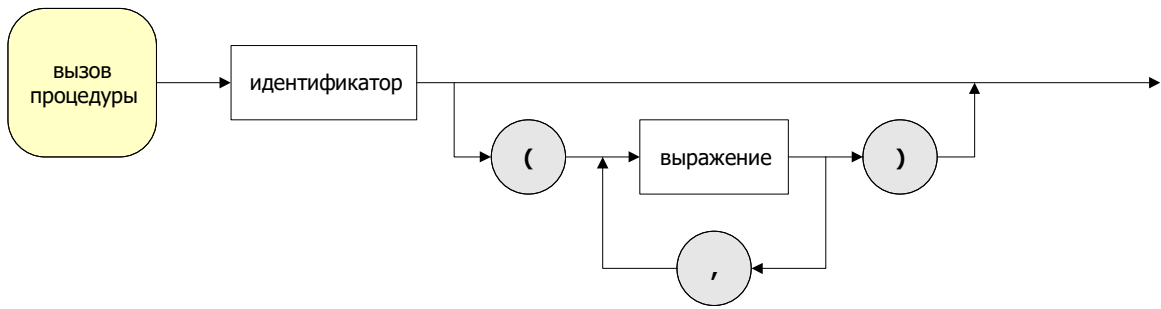


Рисунок 46 - Диаграмма описания вызова процедуры

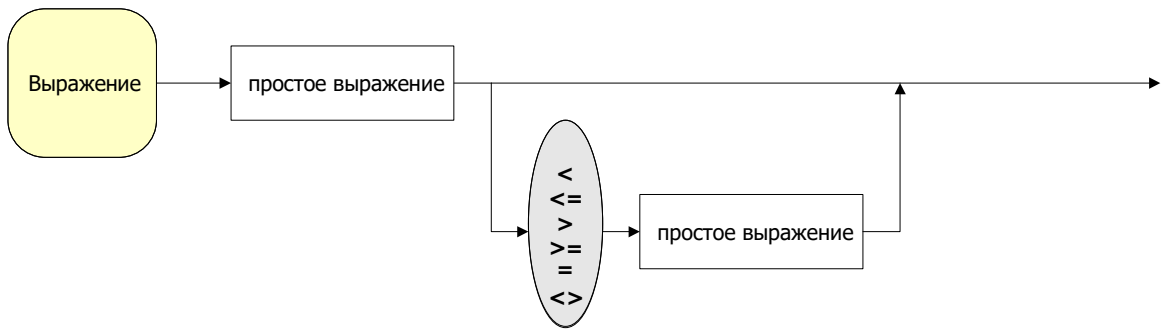


Рисунок 47 - Диаграмма описания выражения

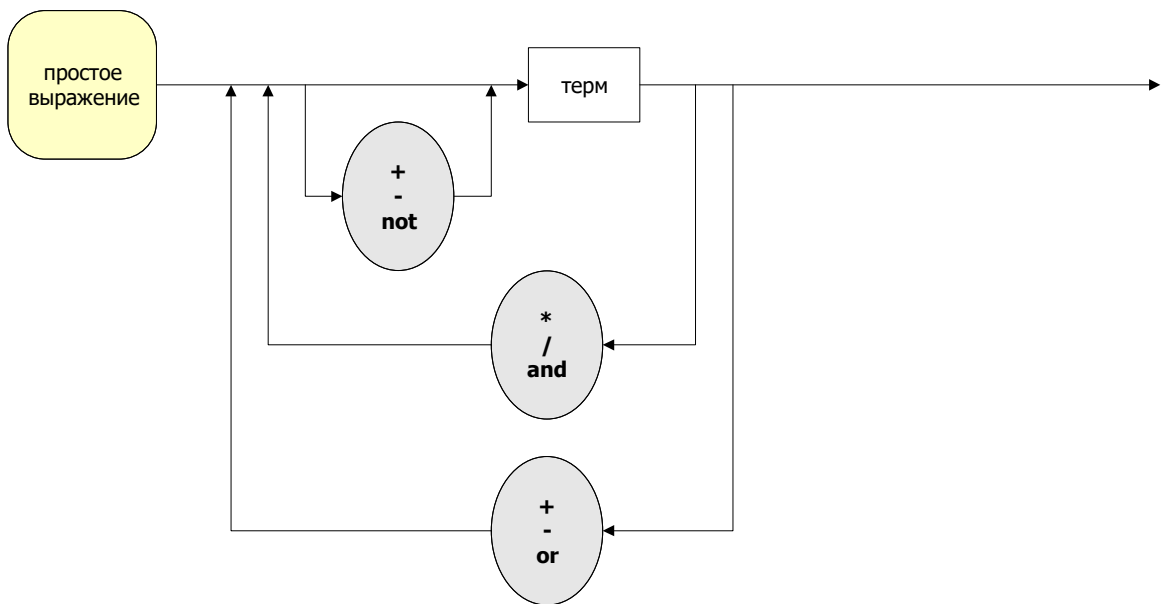


Рисунок 48 - Диаграмма описания простого выражения

Инв. № подл.	Подп. и дата	Взаим. инв. №	Инв. №	Подп. и дата

Изм	Лист	№ докум	Подп.	Дата

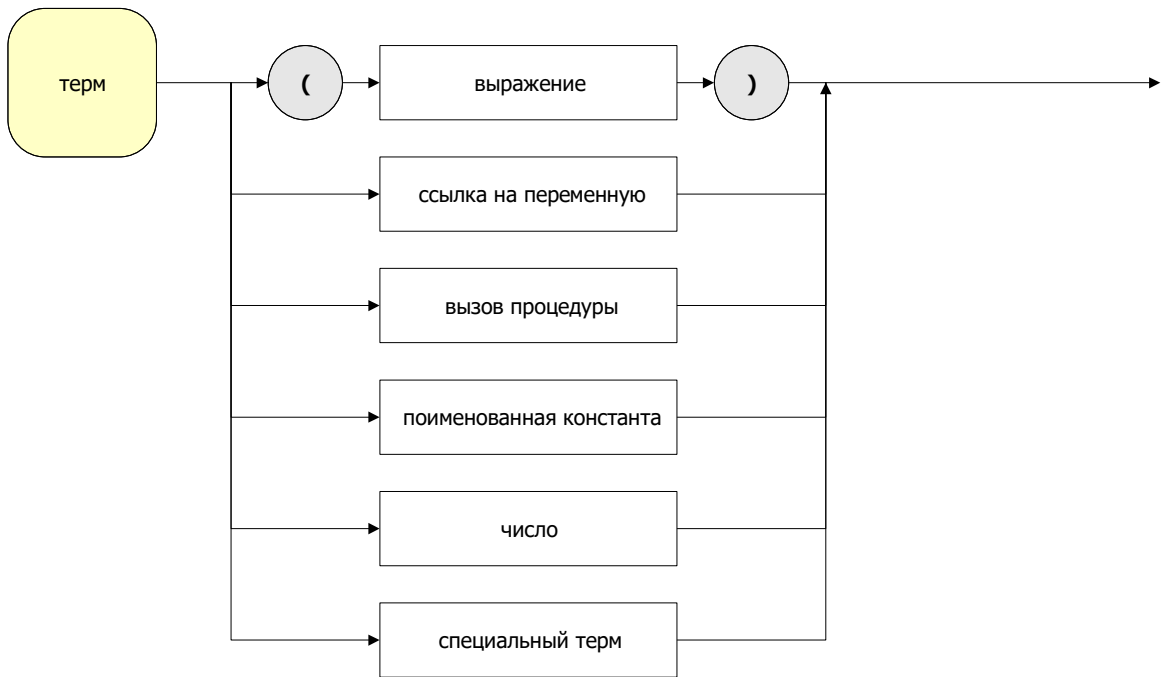


Рисунок 49 - Диаграмма описания термина

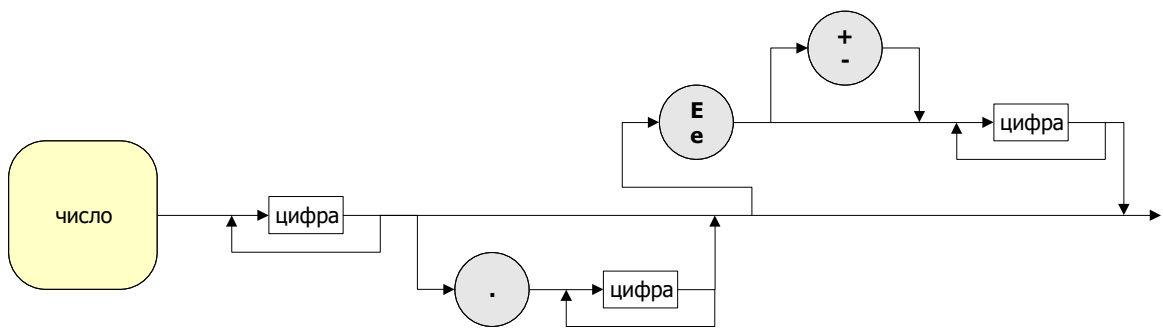


Рисунок 50 - Диаграмма описания числа



Рисунок 51 - Диаграмма описания ссылки на переменную

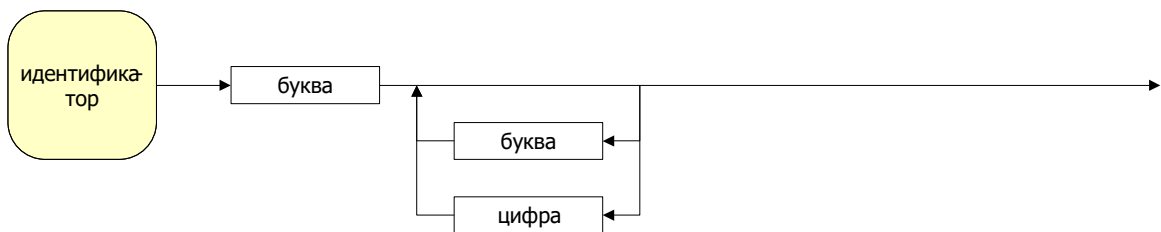


Рисунок 52 - Диаграмма описания идентификатора

Инв. № подл.	Подп. и дата
	Инв. №
Инв. № инв.	Подп. и дата
	Взаим. инв. №
Инв. № докум.	Подп. и дата
	Изм

Изм	Лист	№ докум	Подп.	Дата
-----	------	---------	-------	------

