



# SCADA ZETView. Руководство пользователя.

© 2013 ЗАО"ЭТМС"

ЭТМС.05000-01 34 РО

# Содержание

## SCADA ZETView. Руководство пользователя.

О SCADA ZETView.....	9
Организация данного руководства.....	17
<b>Часть 1. Знакомство со SCADA ZETView.....</b>	<b>18</b>
1. Системные требования .....	18
2. Установка ZETView .....	20
3. Обновление ZETView .....	28
4. Запуск ZETView .....	35
5. Настройка ZETView .....	39
6. Работа с компонентами .....	44
7. Соединение компонентов .....	48
8. Интерфейс оператора и интерфейс проектирования .....	54
9. Свойства компонентов .....	56
10. Настройка внешнего вида, выравнивание и позиционирование .....	60
11. Многостраничные проекты .....	64
12. Запуск, остановка и сохранение проекта. Компиляция в EXE .....	66
<b>Часть 2. Основы проектирования в SCADA ZETView.....</b>	<b>68</b>
1. Введение .....	68
2. Ввод данных, индикация и управление .....	68
3. Расширяем возможности настройки .....	77
4. Работа с текстом .....	80
5. Основы использования массивов .....	83
6. Конвертация .....	86
7. Арифметика и логика .....	89
8. Отладка .....	90
<b>Часть 3. Основы создания проектов в SCADA ZetView.....</b>	<b>91</b>
1. Часть 2. Основы создания проектов в SCADA ZetView .....	91
<b>Часть 4. Компоненты ZETView.....</b>	<b>92</b>
1. Общие свойства (окружение) .....	92
2. Автоматизация .....	93
2.1. ПИД-регулятор.....	93
2.2. ПИД-регулятор из таблицы.....	94
2.3. Синхронизация по GPS.....	97
2.4. Фильтрация.....	98
3. Анализ сигналов .....	100
3.1. Анализ нелинейных искажений.....	100
3.2. Взаимный долектавный спектр.....	101
3.3. Взаимный корреляционный анализ .....	103
3.4. Взаимный узкополосный спектр.....	105
3.5. Высокочастотный осциллограф.....	107
3.6. Гистограмма.....	109
3.7. Детектор землетрясений.....	111
3.8. Детектор STA/LTA.....	115
3.9. Долектавный спектр.....	120



3.10. Модальный анализ.....	124
3.11. Обнаружитель событий.....	127
3.12. Узкополосный спектр.....	128
<b>4. Арифметика .....</b>	<b>131</b>
4.1. Арифметика.....	131
4.2. Арктангенс.....	135
4.3. Вычитание.....	136
4.4. Деление.....	138
4.5. Инкрементирование.....	140
4.6. Интегратор.....	141
4.7. Логарифм.....	143
4.8. Модуль.....	145
4.9. Округление.....	146
4.10. Остаток от деления.....	148
4.11. Попадание в диапазон.....	150
4.12. Разность.....	151
4.13. Сложение.....	153
4.14. Сравнение (неравенство).....	154
4.15. Сравнение (равенство).....	156
4.16. Степень.....	158
4.17. Сумматор.....	160
4.18. Умножение.....	161
4.19. Усреднение.....	163
<b>5. Внешние устройства .....</b>	<b>165</b>
5.1. Акустический модем.....	165
5.2. Весы VP05mc.....	166
5.3. Генератор DS360.....	167
5.4. Источник питания LPS-305.....	169
5.5. Источник питания MCA 750-3000.....	170
5.6. Источник питания PPE-3323.....	172
5.7. Источник питания PSH-3610.....	173
5.8. Источник питания PSM-2010.....	174
5.9. Источник питания Б5-85/1.....	175
5.10. Мультиметр Agilent 34401a.....	176
5.11. Нановольтметр Keithley 2182a.....	177
<b>6. Время .....</b>	<b>178</b>
6.1. Временная задержка.....	178
6.2. Время сервера.....	180
6.3. Информация о дате.....	181
6.4. Таймер.....	183
<b>7. Вход (АЦП) .....</b>	<b>183</b>
7.1. Виртуальный канал.....	183
7.2. Измерительный канал.....	185
7.3. Канал формулы.....	187
7.4. Опрос параметров канала.....	189
7.5. Список каналов.....	192
7.6. Фиксация изменений.....	194
<b>8. Выбор .....</b>	<b>197</b>
8.1. Вертикальная прокрутка.....	197
8.2. Горизонтальная прокрутка.....	198
8.3. Демультимплексор.....	200
8.4. Календарь.....	202
8.5. Комбинированный список.....	203
8.6. Константа.....	205

8.7.Мультиплексор.....	207
8.8.Переменная.....	210
8.9.Селектор № 1.....	213
8.10.Селектор № 2.....	215
8.11.Селектор № 3.....	217
8.12.Селектор № 4.....	218
8.13.Список.....	220
<b>9.Генератор (ЦАП) .....</b>	<b>222</b>
9.1.Амплитудная модуляция.....	222
9.2.Воспроизведение из файла.....	225
9.3.Входной канал.....	227
9.4.Импульсный сигнал.....	229
9.5.Линейная частотная модуляция.....	231
9.6.Логарифмическая частотная модуляция.....	235
9.7.Пилообразный сигнал.....	239
9.8.Радиоимпульсный сигнал.....	244
9.9.Синусоидальный сигнал.....	247
9.10.Синхронный генератор.....	250
9.11.Суперпозиция генераторов.....	251
9.12.Частотная модуляция.....	257
9.13.Шум.....	262
<b>10.Группы .....</b>	<b>266</b>
10.1.Объединить в группу.....	267
10.2.Панель Windows стиль.....	268
<b>11.Измерение (ВП) .....</b>	<b>271</b>
11.1.Виброметр.....	271
11.2.Вольтметр переменного тока.....	284
11.3.Вольтметр постоянного тока.....	288
11.4.Диапазон значений.....	290
11.5.Мгновенное значение.....	292
11.6.Параметры дисбаланса.....	295
11.7.Селективный вольтметр.....	296
11.8.Тахометр.....	299
11.9.Текущий уровень.....	300
11.10.Тензометр.....	302
11.11.Термометр ТП.....	311
11.12.Термометр ТС.....	317
11.13.Фазометр.....	322
11.14.Частотомер.....	326
11.15.Энкодер.....	327
<b>12.Индикация .....</b>	<b>346</b>
12.1.Вертикальный индикатор процесса.....	346
12.2.Горизонтальный индикатор процесса.....	347
12.3.ЖК индикатор.....	348
12.4.Индикатор процесса.....	349
12.5.Индикатор уровня.....	351
12.6.Колба.....	352
12.7.Рисунок.....	354
12.8.Световой индикатор.....	356
12.9.Стрелочный индикатор.....	357
12.10.Стрелочный индикатор №2.....	358
12.11.Термометр.....	360
12.12.Цифровой индикатор.....	362
<b>13.Кнопки .....</b>	<b>363</b>

13.1. CheckBox .....	363
13.2. Кнопка с картинкой .....	364
13.3. Кнопка с фиксацией .....	366
13.4. Кнопка без фиксации .....	367
13.5. Тумблер .....	368
<b>14. Конвертация .....</b>	<b>370</b>
14.1. Конвертер (логика - строка) .....	370
14.2. Конвертер (логика - число) .....	372
14.3. Конвертер (стр.мас. - строка) .....	373
14.4. Конвертер (стр.мас. - числ.мас.) .....	375
14.5. Конвертер (строка - число) .....	377
14.6. Конвертер (числ.мас.-стр.мас.) .....	378
14.7. Конвертер (число - строка) .....	379
14.8. Преобразование текста в речь .....	381
14.9. Преобразование цвета .....	382
<b>15. Логика .....</b>	<b>384</b>
15.1. D-триггер .....	384
15.2. RS-триггер .....	386
15.3. Битовая маска .....	390
15.4. Двоичное число .....	392
15.5. Логическое И .....	396
15.6. Логическое И (множественное) .....	399
15.7. Логическое ИЛИ .....	400
15.8. Логическое ИЛИ (множественное) .....	403
15.9. Логическое Не .....	404
15.10. Наложение маски .....	406
<b>16. Массивы .....</b>	<b>409</b>
16.1. Детерминированный массив .....	409
16.2. Детерминированный массив из архива .....	411
16.3. Запись массива .....	412
16.4. Матрица .....	414
16.5. Формировщик массивов (дата-массив) .....	416
16.6. Формировщик массивов (деление строки) .....	417
16.7. Формировщик массивов (массив-массив) .....	418
16.8. Формировщик массивов (текст-массив) .....	422
16.9. Формировщик массивов (число-массив) .....	424
<b>17. Массив - Значение .....</b>	<b>426</b>
17.1. Значение элемента (строковый) .....	426
17.2. Значение элемента (числовой) .....	428
17.3. Индекс превышения .....	430
17.4. Максимальное значение массива .....	432
17.5. Минимальное значение массива .....	434
17.6. Разброс значений .....	436
17.7. Размер массива (строковый) .....	437
17.8. Размер массива (числовой) .....	439
17.9. Синхронизация массива .....	440
17.10. Среднее арифметическое значение .....	441
17.11. Среднеквадратичное значение .....	444
17.12. Среднеквадратичное отклонение .....	446
17.13. Сумма элементов массива .....	448
<b>18. Массив-Массив .....</b>	<b>450</b>
18.1. Взятие модуля .....	450
18.2. Гистограмма .....	452
18.3. Дифференцирование массива .....	455

18.4.Инверсия массива.....	455
18.5.Интегрирование массива.....	457
18.6.Медианный фильтр.....	459
18.7.Метод наименьших квадратов.....	460
18.8.Нормализация массива.....	461
18.9.Обратное преобразование Фурье.....	462
18.10.Огибающая.....	462
18.11.Ограничение массива.....	463
18.12.Операция с константой.....	468
18.13.Преобразование координат в декартовы.....	470
18.14.Преобразование координат в полярные.....	471
18.15.Преобразование Фурье.....	472
18.16.Ударный спектр.....	474
18.17.Расчет уровня в дБ.....	476
18.18.Сокращение массива.....	477
18.19.Сортировка массива.....	479
18.20.Фильтрация массива.....	480
<b>19.Отображение .....</b>	<b>481</b>
19.1.XYZ осциллограф.....	481
19.2.XYZ плоттер.....	484
19.3.Грамма.....	486
19.4.График в полярных координатах.....	488
19.5.График.....	490
19.6.Карта.....	492
19.7.Многоканальный осциллограф.....	494
19.8.Таблица данных.....	496
19.9.3D-моделирование.....	497
<b>20.Программируемая логика .....</b>	<b>498</b>
20.1.Программируемый компонент.....	498
20.2.Список переменных к сценарию.....	504
20.3.Сценарий.....	505
20.4.Цикл for.....	508
20.5.Цикл w hile.....	510
<b>21.Потоковая обработка .....</b>	<b>511</b>
21.1.Арифмометр.....	511
21.2.Компаратор.....	513
21.3.Пиковый обнаружитель.....	515
21.4.Синхронизатор (1 канал).....	516
21.5.Синхронизатор (4 канала).....	517
<b>22.Протоколы обмена и шины данных .....</b>	<b>518</b>
22.1.Интерфейс RS-232.....	518
<b>23.Разное .....</b>	<b>519</b>
23.1.Выбор цвета.....	519
23.2.Выход из программы.....	522
23.3.Геометрическая фигура.....	523
23.4.Инициализатор.....	526
23.5.Концентратор.....	528
23.6.Межстраничная связь.....	529
23.7.Ключ.....	533
23.8.Менеджер свойств.....	534
23.9.Менеджер стандартных свойств.....	536
23.10.Переход на страницу.....	538
23.11.Прослушивание канала.....	539
23.12.Случайное число.....	540

<b>24.Регистратор</b> .....	<b>542</b>
24.1.Автономный регистратор.....	542
24.2.Запись каналов.....	543
<b>25.Решения</b> .....	<b>544</b>
25.1.Обнаружение близких СВ.....	544
25.2.Пеленг.....	545
<b>26.Сообщения оператору</b> .....	<b>546</b>
26.1.Журнал событий.....	546
26.2.Звуковая сигнализация.....	547
26.3.Маркер качества (к журналу событий).....	548
26.4.Расширенный журнал.....	549
26.5.Сообщение.....	552
<b>27.Текст</b> .....	<b>554</b>
27.1.Метка.....	554
27.2.Поиск слова в строке.....	556
27.3.Сложение строк.....	558
27.4.Статический текст Window s стиль.....	560
27.5.Текстовое поле.....	561
27.6.Разделитель строки.....	562
<b>28.Устройства ZET</b> .....	<b>564</b>
28.1.Коммутационный блок.....	564
28.2.ZET 0xxx2.....	565
28.3.ZET 0xxx4/0xxx8.....	566
28.4.ZET 110.....	567
28.5.ZET 210.....	569
28.6.ZET 220.....	570
28.7.ZET 230.....	571
28.8.ZET 240.....	572
28.9.Синхронный генератор ZET7090.....	573
28.10.Смена адреса ZET7xxx.....	574
28.11.Событийный канал ZETxxx.....	575
28.12.Тензодатчик ZET7010.....	576
28.13.Термометр термопары ZET7020.....	577
28.14.Термометр термосопротивления ZET7021.....	579
28.15.Управление реле ZET7062.....	580
28.16.Устройство ZET.....	581
28.17.Цифровой генератор ZET7060g.....	582
28.18.Энкодер ZET7060E.....	583
<b>29.Файл</b> .....	<b>584</b>
29.1.Базы данных.....	584
29.2.Выбор каталога.....	591
29.3.Выбор файла.....	592
29.4.Запись в файл.....	595
29.5.Запуск программы по имени.....	595
29.6.Отчет.....	597
29.7.Поиск файлов в директории.....	600
29.8.Пути ZETLab.....	602
29.9.Точка доступа.....	605
29.10.Чтение из файла.....	607
<b>30.Цифровой порт</b> .....	<b>609</b>
30.1.Цифровой порт.....	609
<b>31.OPC</b> .....	<b>612</b>
31.1.Исторические данные.....	612
31.2.Обмен данными с OPC.....	613

<b>32. Excel</b> .....	<b>615</b>
32.1. Документ Excel.....	615
32.2. Страница Excel.....	618
32.3. Ячейка Excel.....	620

## О SCADA ZETView

ZETView это мощная среда разработки автоматизированных рабочих мест на базе ПК с использованием программной и аппаратной измерительной части. Возможно, пользователю будет интересно узнать о возможностях этой системы. Главная особенность SCADA ZETView в том, что фактически она является средой программирования, а следовательно, позволяет решать множество задач, таких как создание уникальных виртуальных приборов, измерительных комплексов, автоматизированных испытательных стендов, систем мониторинга и управления и т.д.

Без использования специальных инструментов, решение большинства задач занимает длительное время. Например, при тестировании оборудования и снятии характеристик по множеству каналов. Вручную – это огромный объем работы, требующий много сил и внимания. Автоматизация процесса посредством написания программы на одном из стандартных языков программирования, требует высокой квалификации и занимает много времени. Мощная графическая среда разработки SCADA ZETView специально предназначена для создания приложений, применяющихся в различных областях промышленности, значительно увеличит производительность труда.

Сочетая в себе простоту и удобство графической системы проектирования и гибкость языка программирования ZETView открывает пользователю более простые и удобные пути решения целого ряда инженерных проблем.

SCADA ZETView создана для облегчения работы по решению ваших задач. Для этого система оснащена большим набором компонентов, при помощи которых, как из кубиков, можно строить проекты любой сложности. ZETView содержит в себе библиотеки различных виртуальных приборов, осуществляющих измерения и анализ данных, поступающих с подключаемых к компьютеру аппаратных средств или использующих записанный ранее сигнал. Компоненты обладают множеством свойств, которые могут быть заданы при создании проекта как постоянные или настраиваемые оператором в процессе работы проекта.

В ZETView реализовано множество средств отображения и индикации, таких как, различные графики, в том числе в 3D, таблицы или индикаторы: стрелочные, цифровые, световые, индикаторы процесса. Все компоненты индикации обладают настраиваемыми параметрами отображения. Возможность управления проектами, используя графические элементы управления и индикации, упрощает их использование и восприятие. На многих предприятиях введены стандарты представления и оформления информации, используя SCADA ZETView, вы сможете представлять результаты в требуемой форме.

SCADA ZETView идеально подходит для создания полностью



автоматизированных проектов, самостоятельно осуществляющих измерения, анализ, управление и оформление результатов. Поэтому вам достаточно просто запустить проект, а все остальное можно поручить технике.

### **Программирование в ZETView**

Разработка приложений в SCADA ZETView принципиально отличается от разработки с использованием традиционных языков программирования. В любом из известных языков программирование основано на вводе команд и параметров, образующих программный код, в ZETView вместо текстовых команд используются графические компоненты. Взаимодействие между компонентами организуется путем проведения связей, связанные компоненты образуют схему. Таким образом, SCADA ZETView избавит вас от рутинной работы по созданию корректного кода, работы с указателями, распределением памяти и прочими «радостями» в работе программиста. Вы сможете сконцентрировать все свое внимание на реализации алгоритма. Кроме того, отладка и поиск ошибок, при графическом представлении кода, станет на порядок проще. ZETView имеет два режима отображения: интерфейс проектирования и интерфейс оператора. Таким образом, человеку, использующему ваш проект, не нужно вдаваться в подробности его работы. Особенность работы со SCADA ZETView, это ее ориентированность на работу с непрерывным потоком данных, по этому принципу построена работа большинства компонентов, так же данный подход следует учитывать при разработке алгоритмов. Графический язык разработки SCADA ZETView интуитивно понятен, даже пользователям ранее не сталкивающимся с программированием и системами разработки будет не сложно освоить ее.

Схема – отражает структурную схему измерений, т.е. если нарисовать блок-схему измерительного процесса, останется только воссоздать ее в среде графического проектирования ZETView. Таким образом, для решения задачи нужно лишь ее сформулировать.

Связи – отражают передачу данных, причинно-следственные связи, последовательность действий

Пиктограммы, отражающие назначение, позволяют не читать название, а воспринимать компонент сразу, а визуальное восприятие ускоряет процесс понимания.

Программа в виде мнемосхемы, в которой назначение каждого компонента очевидно, гораздо более понятна для понимания и «читается с листа».

SCADA ZETView неразрывно связана с комплексом программ ZETLab. ZETLab - это набор виртуальных приборов, которые интегрированы в SCADA ZETView.

Фактически ZETView является оболочкой для ZETLab.



Рассмотрим работу ZETView на примере, в котором оцениваются динамические характеристики строительной конструкции. Вблизи несущих опор здания устанавливаются сейсмодатчики воспринимающие колебания конструкции и передающие сигнал на сейсмоприемники. Сигналы с сейсмоприемников оцифровываются сеймостанцией и передаются на ПК в режиме реального времени по сети Ethernet. С помощью драйвера данные попадают на программный сервер каналов ZET Server. Поток данных с сервера каналов поступает на виртуальный прибор

Как видно из рисунка, SCADA ZETView имеет множественный доступ к приборам ZETLab, это означает, что если в вашем проекте используется, например, пять вольтметров, то ZETView запустит пять копий программы «Вольтметр» из пакета ZETLab, для каждой копии выделяется собственная область памяти. Это обеспечивает работу проекта, даже если, например, по каким-либо причинам пропадет сигнал с одного из вольтметров.

## Демонстрационные примеры

### *Система мониторинга конструкций зданий*

Система мониторинга конструкций зданий представляет собой аппаратно-программный диагностический комплекс для экспериментального определения основных динамических характеристик строительных конструкций.

#### Аппаратная часть

Аппаратная часть системы мониторинга конструкций зданий реализуется на базе сеймостанции ZET048 в промышленном исполнении и сейсмоприемников BC1313.

Сейсмоприемники BC1313 являются трехкомпонентными акселерометрами и регистрируют колебаний здания по трем взаимноперпендикулярным осям X, Y и Z. Сейсмоприемники BC1313 устанавливают вблизи несущих опор зданий таким образом, чтобы измерительная ось Z была направлена вертикально вверх. При этом по оси Z будут регистрироваться колебания здания, связанные с сейсмической активностью Земли, а по осям X и Y - колебания, связанные с порывами ветра.

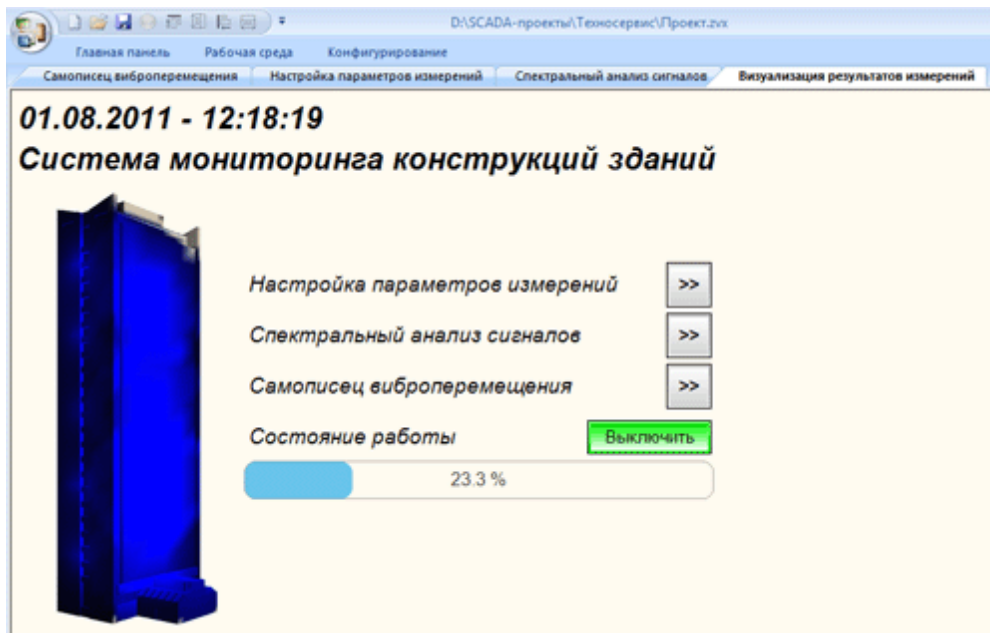
Сигналы с сейсмоприемников оцифровываются сейсмостанцией и передаются на ПК в режиме реального времени по сети Ethernet. Сейсмостанция также обеспечивает непрерывный контроль питания датчиков и их периодическую поверку без демонтажа.

Программная часть

Программная часть комплекса выполнена в среде графического программирования ZETView и представляет собой SCADA-проект (исходный код в виде мнемосхемы) и исполнительный файл.

Программа состоит из четырех окон:

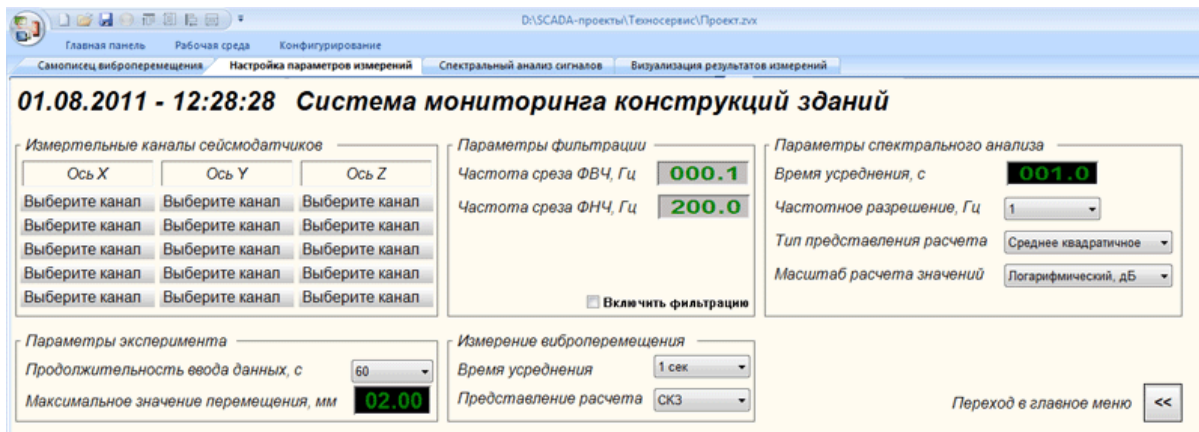
1) Визуализация результатов измерений:



Окно «Визуализация результатов измерений»

Окно визуализации результатов измерений предназначено для отображения модели исследуемого здания, а также для старта или окончания процесса измерений. Во время проведения эксперимента, в зависимости от интенсивности воздействия, воспринимаемого установленными на здании датчиками, модель окрашивается в различные цвета. Цвет модели пропорционален воздействию (в частности, виброперемещению) в точке крепления датчика (кроме виброперемещения, настройки программы позволяет контролировать виброскорость и виброускорение). Запуск и остановка эксперимента осуществляется по нажатию на соответствующую кнопку. По нажатию на кнопки «Настройка параметров измерений», «Спектральный анализ сигналов» и «Самописец виброперемещения» происходит переход в окна программы с одноименным названием.

## 2) Настройка параметров измерений:



Окно «Настройка параметров измерений»

Данное окно предназначено для настройки параметров измерений.

В блоке «Измерительные каналы сейсмодатчиков» задается соответствие измерительных каналов системы реальным сигналам. Для установки любого канала достаточно нажать на его поле манипулятором и выбрать из появившегося списка нужный сигнал.

В блоке «Параметры фильтрации» задаются настройки полосового фильтра. При помощи элемента включения «Включить фильтрацию» можно включить или выключить полосовую фильтрацию сигналов. В полях «Частота среза ФВЧ, Гц» и «Частота среза ФНЧ, Гц» задаются частоты среза фильтра высоких и низких частот соответственно.

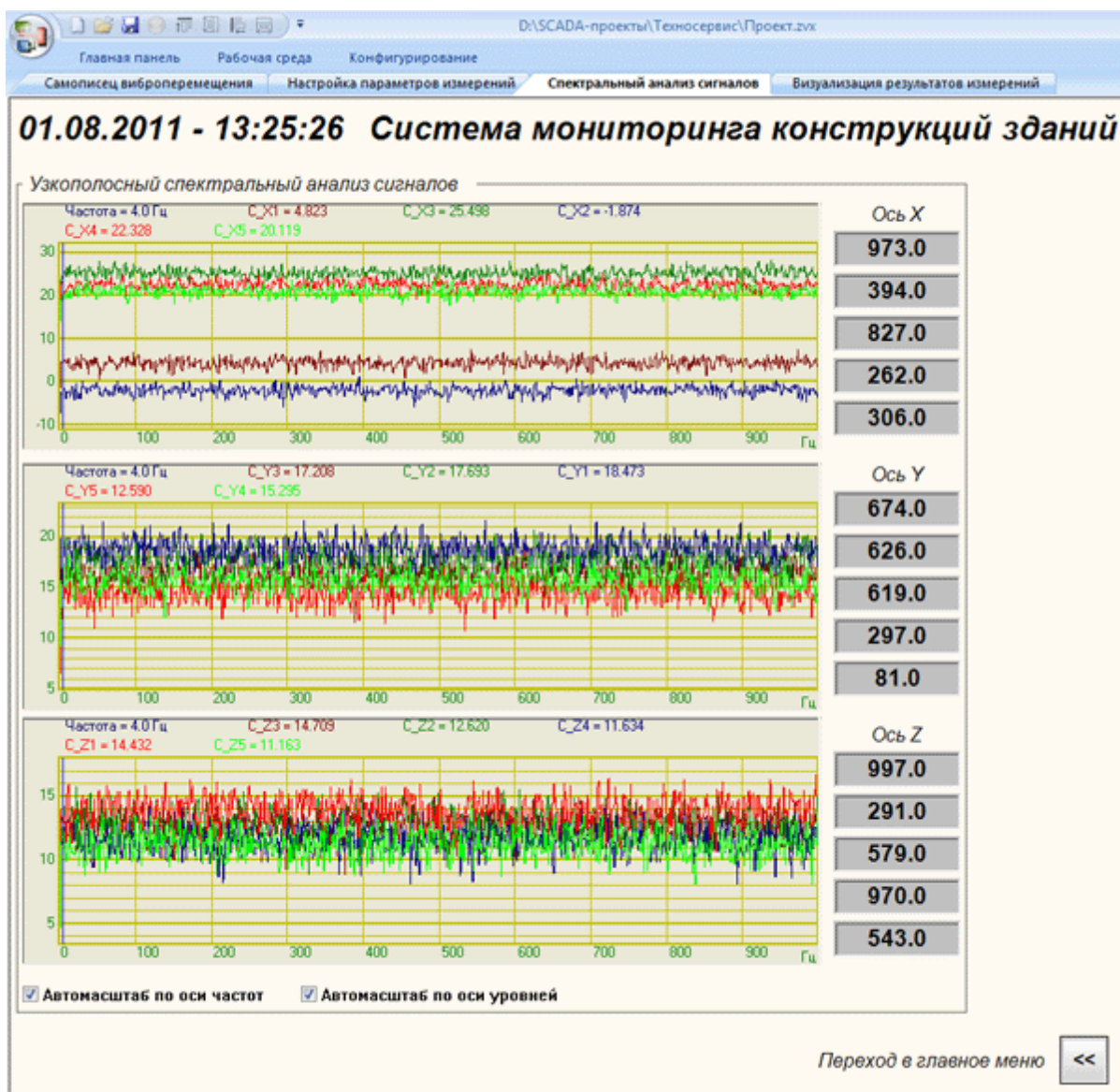
В блоке «Параметры спектрального анализа» задаются настройки анализатора спектра сигналов. Время усреднения мгновенного спектра задается при помощи селектора «Время усреднения, с». Элемент выбора «Частотное разрешение, Гц» дает возможность выбрать текущее частотное разрешение анализатора. Элемент «Тип представления расчета» дает возможность представлять результаты расчетов одним из способов: спектральная плотность, спектральная мощность, среднее квадратичное и пиковое значение. Элемент «Масштаб расчета значений» предназначен для выбора отображения результатов расчета спектра в логарифмическом масштабе (относительно опорного значения для вычисления дБ) или линейном масштабе (в единицах измерения).

Блок «Параметры эксперимента» содержит два элемента: «Продолжительность ввода данных, с» и «Максимальное значение перемещения, мм». Первый элемент отвечает за длительность эксперимента, задает время записи сигналов в течение эксперимента, а также устанавливает время усреднения накопленных значений мгновенного спектра

сигналов. Второй элемент устанавливает значение, относительно которого рассчитываются цвета раскраски модели здания.

В блоке «Измерение виброперемещения» происходит настройка виброметра, а именно устанавливается время усреднения при расчете (элемент «Время усреднения») и тип представления расчета (СКЗ, пиковое или амплитудное значение) в элементе «Представление расчета».

3) Спектральный анализ сигналов:

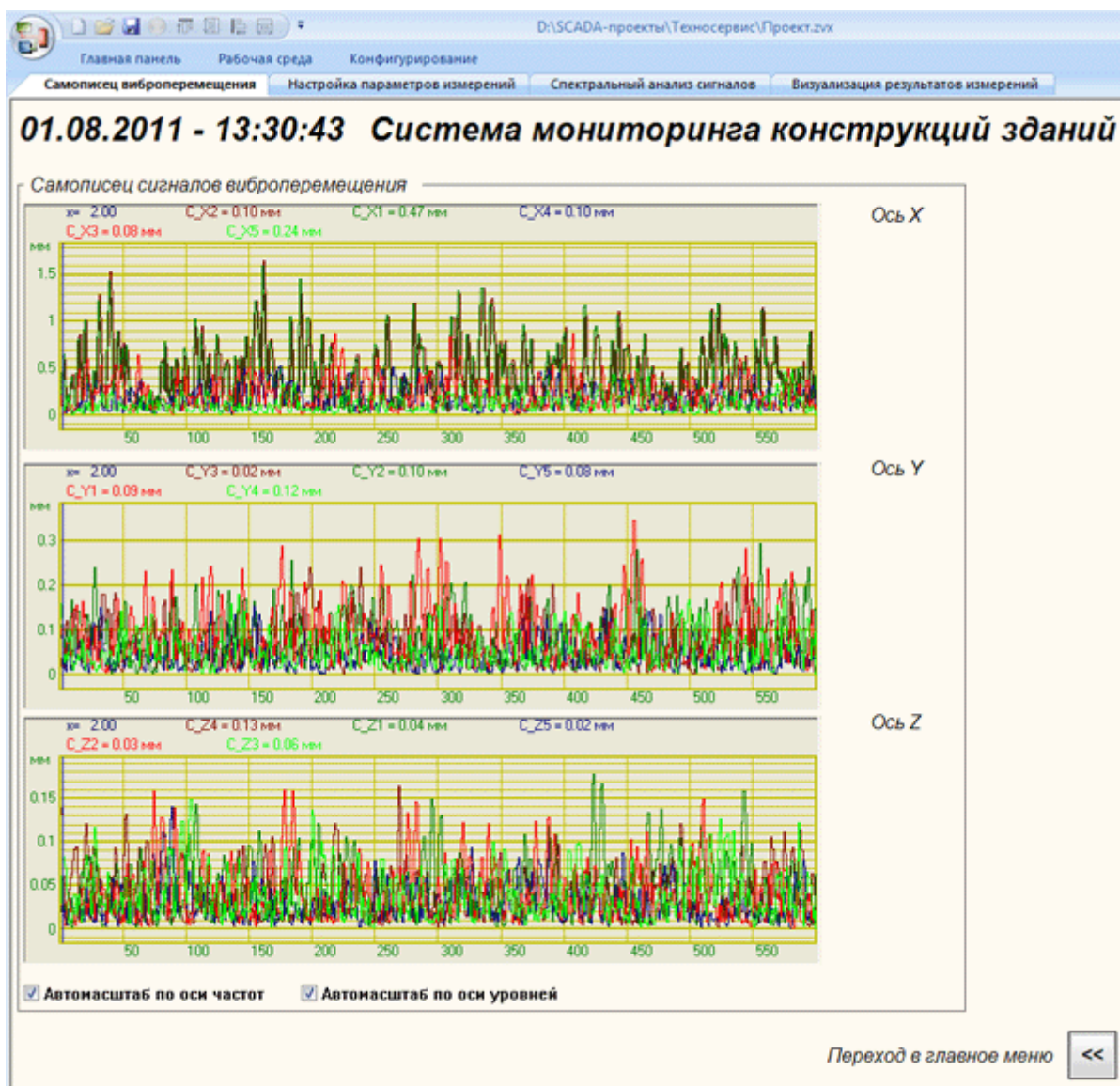


Окно «Спектральный анализ сигналов»

Данное окно предназначено для отображения результатов спектрального анализа

сигналов. На трех графиках представлены спектры составляющих X, Y и Z сигналов сейсмодатчиков, а также частота максимального значения спектра по каждому датчику.

4) Самописец виброперемещения:



Окно «Самописец виброперемещений»

Данное окно предназначено для отображения в ходе эксперимента измеренных значения виброперемещения по каждой из составляющих каждого датчика.





# Организация данного руководства

Данное руководство состоит из четырех частей.

В первой части читатель получит общие начальные знания для работы со SCADA ZETView. Здесь описано как установить, настроить SACAD ZETView и начать работу.

Во второй части внимание уделяется компонентам. Несмотря на то, что их количество постоянно растет, освоив вторую часть, пользователь будет знать, какие есть основные компоненты, научится использовать виртуальные приборы, поймет их логику работы с потоками данных, будет знать какие компоненты используются для решения определенных задач, что поможет ему самостоятельно разобраться с другими компонентами.

Третья часть представляет собой набор сложных проектов, с описанием логики и особенностей работы, советами по использованию тех или иных блоков и компонентов для решения задачи. ( На данный момент третья глава находится в стадии разработки и не включена в руководство.)

Четвертая глава представляет собой справочник по компонентам. В описании каждого компонента содержится простой пример, показывающий его работу. Примеры представлены в формате \*.zvx, поэтому при желании пользователь сможет поэкспериментировать с готовыми проектами.

# Часть 1. Знакомство со SCADA ZETView

## Глава 1. Системные требования

Программное обеспечение **ZETLab** предназначено для использования на персональных компьютерах типа IBM PC Intel® Pentium®/Celeron®/ или совместимые с ними, работающих под управлением русскоязычной (локализованной) либо корректно русифицированной версии операционных систем:

1. Microsoft® Windows® XP с пакетом обновления не ниже SP3.
2. Microsoft® Windows® Vista с пакетом обновления SP1.
3. Microsoft® Windows® 7 32 разрядная с пакетом обновления SP1.
4. Microsoft® Windows® 7 64 разрядная с пакетом обновления SP1.
5. Microsoft® Windows® Server 2003.
6. Microsoft® Windows® Server 2008 32 разрядная
7. Microsoft® Windows® Server 2008 64 разрядная с пакетом обновления SP2.
8. Microsoft® Windows® Server 2008 R2 с пакетом обновления SP1.
9. Microsoft® Windows® Starter (без ограничения на количество запущенных программ).

### Конфигурация компьютера для установки и запуска программного обеспечения ZETLab и драйверов устройств:

- двухядерный процессор или более;
- тактовая частота процессора – не менее 1,6 ГГц;
- наличие интерфейса HighSpeed USB 2.0\*;
- оперативная память – не менее 2 Гб;
- свободное место на жестком диске – не менее 20 Гб;
- видеокарта с 3D-графическим ускорителем, поддержкой OpenGL, DirectX, не менее 128 Мб памяти;
- разрешение экрана не менее 1280×1024;
- наличие манипулятора «мышь» или иного указательного устройства (сенсорный экран, трекбол (track ball), тачпад (TouchPad), графический планшет);
- наличие стандартной клавиатуры или иного устройства ввода (сенсорный экран, графический планшет);
- привод CD-ROM для установки программ.

*\*подключение через интерфейс SuperSpeed USB 3.0 не гарантируется*



## Глава 2. Установка ZETView

Перед установкой SCADA ZETView необходимо убедиться, что на компьютер установлено программное обеспечение ZETLab. Информация по установке ZETLab приведена в руководстве «Программное обеспечение ZETLab. Анализатор спектра. Руководство оператора (часть 1)».

Установка ZETView производится с установочного диска, поставляемого с приобретаемым ZET-устройством или, при обновлении ZETView, с помощью файла «ZETView.msi».

После запуска программы установки, появится следующее окно:

При установке ZETView с установочного диска необходимо вставить диск в дисковод компьютера. При этом запустится программа установки программного обеспечения ZETLab и ZETView. В главном окне программы установки программного обеспечения ZETLab и ZETView (рисунок 1.3.1) необходимо выбрать пункт «Установка программного обеспечения», при этом программа предложит установить ZETLab или ZETView (рисунок 1.3.2) – для установки ZETView необходимо выбрать «Установка ZETView». При этом запустится программа установки ZETView (рисунок 1.3.3).

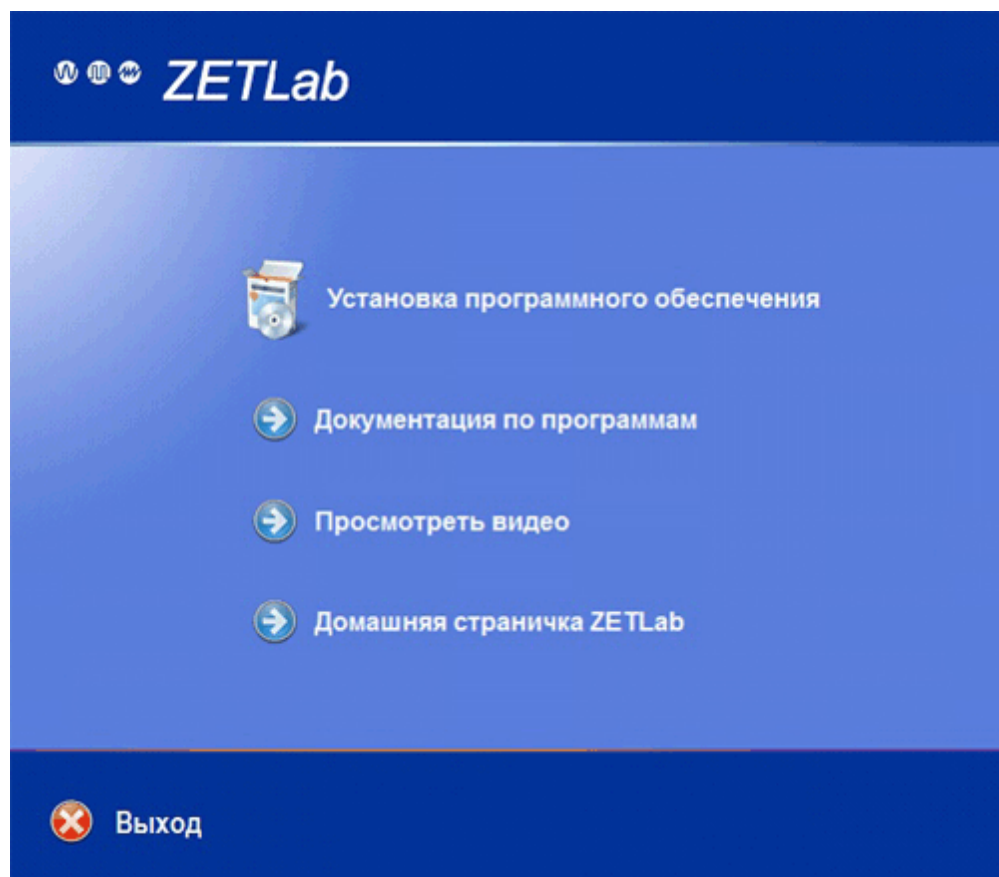


Рисунок 1.3.1

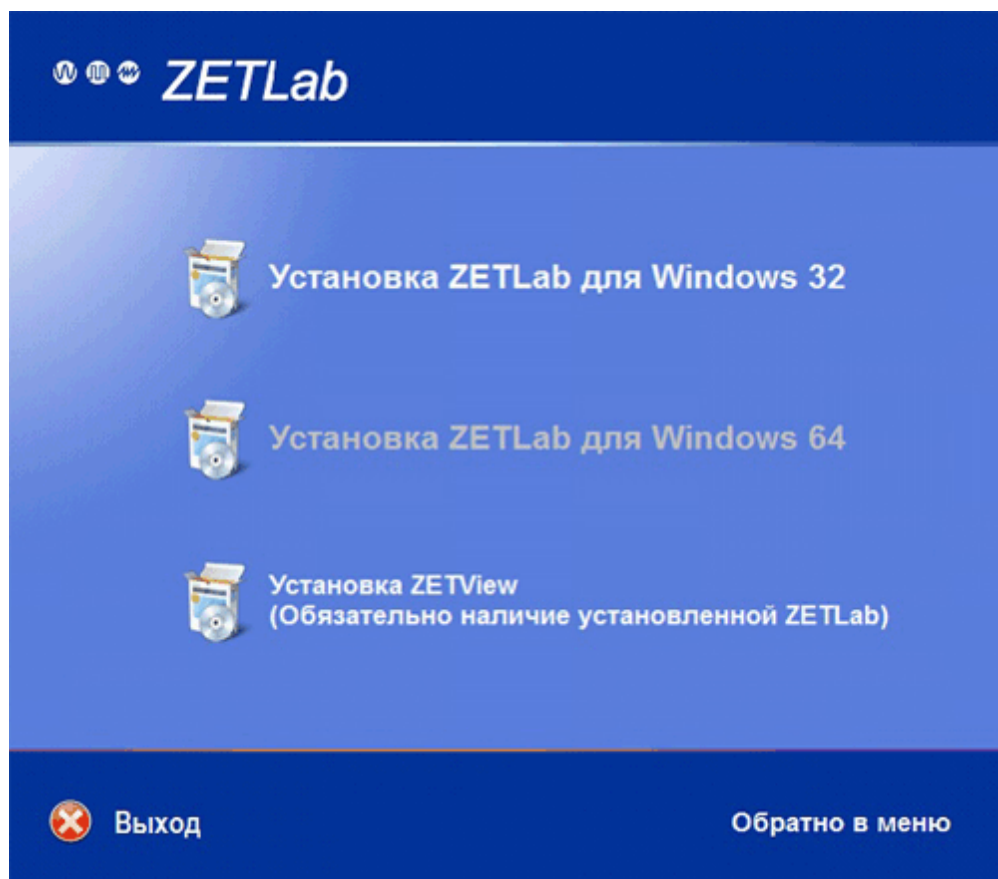


Рисунок 1.3.2

При установке ZETView с помощью файла «ZETView.msi» необходимо запустить файл двойным кликом «мыши», при этом запустится программа установки ZETView (рисунок 1.3.3).

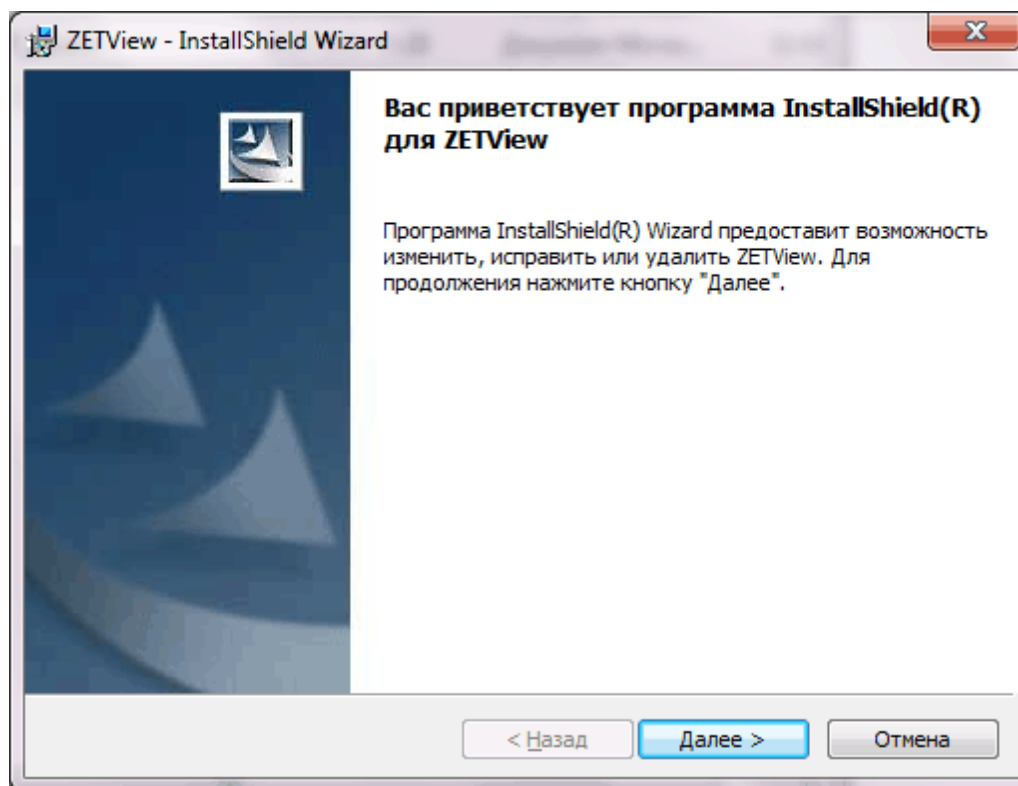


Рисунок 1.3.3. Установка ZETView

При запуске программы установки ZETView для установки ZETView необходимо нажать кнопку «Далее», при этом программа запросит подтверждение установки (рисунок 1.3.4) – для подтверждения установки необходимо нажать кнопку «Далее».



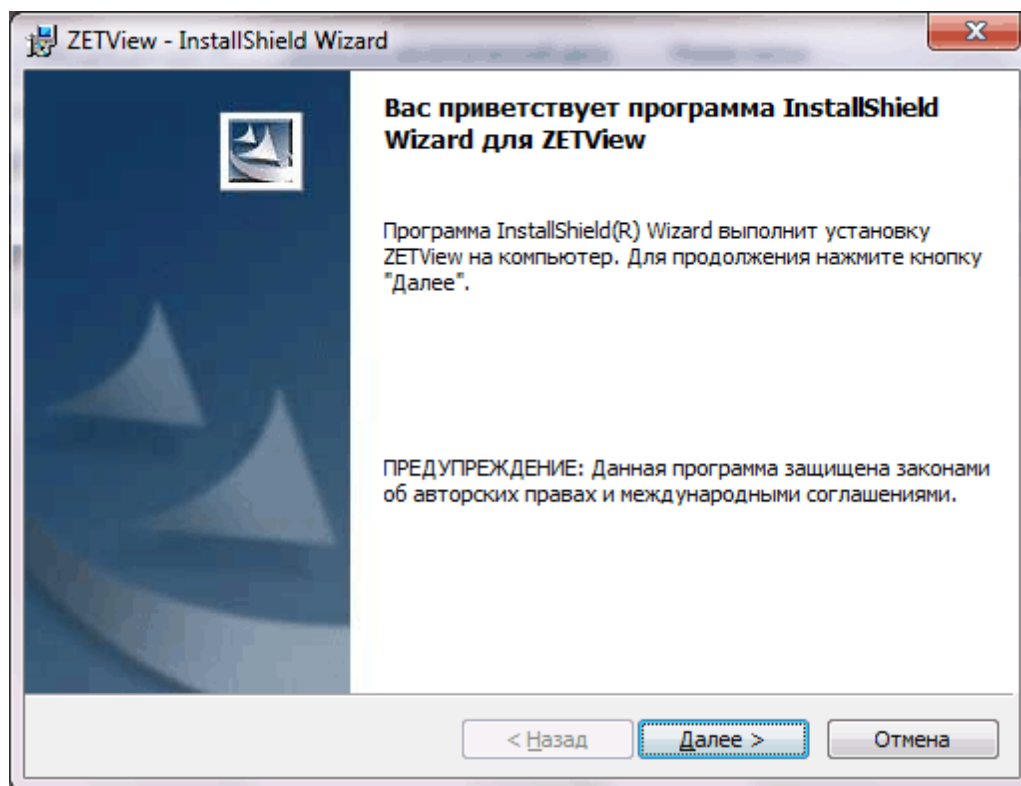


Рисунок 1.3.4.Подтверждеие установки

Далее программа установки ZETView предложит ознакомиться с лицензионным соглашением (рисунок 1.3.5). Лицензионное соглашение можно прочитать в окне программы установки или распечатать его, нажав кнопку «Печать». Без принятия условий лицензионного соглашения установка ZETView будет прервана. При согласии с условиями соглашения необходимо поставить флажок напротив пункта «Я принимаю условия лицензионного соглашения» и нажать кнопку «Далее» для продолжения установки ZETView.

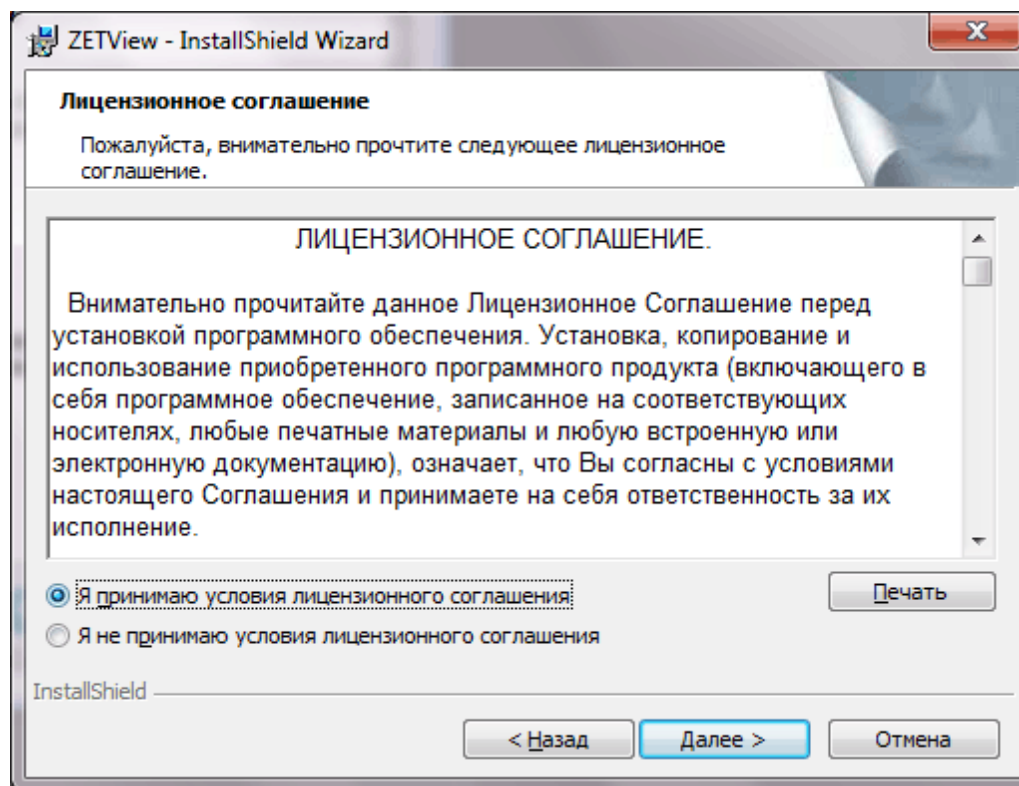


Рисунок 1.3.5.Лицензионное соглашение

Далее запустится установка SCADA ZETView (рисунок 1.3.6). Дождитесь окончания установки.

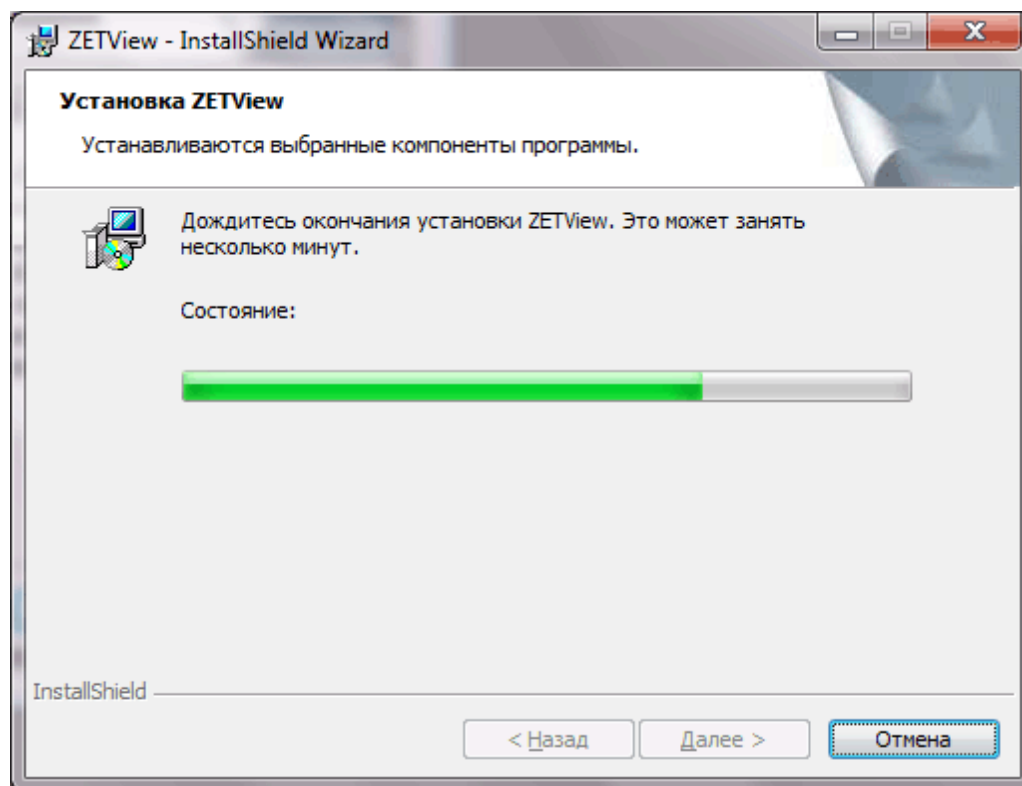


Рисунок 1.3.6. Процесс установки ZETView

По окончании установки программа выдаст сообщение об успешном завершении установки (Рисунок 1.3.7). Для закрытия окна программы установки необходимо нажать кнопку «Готово».

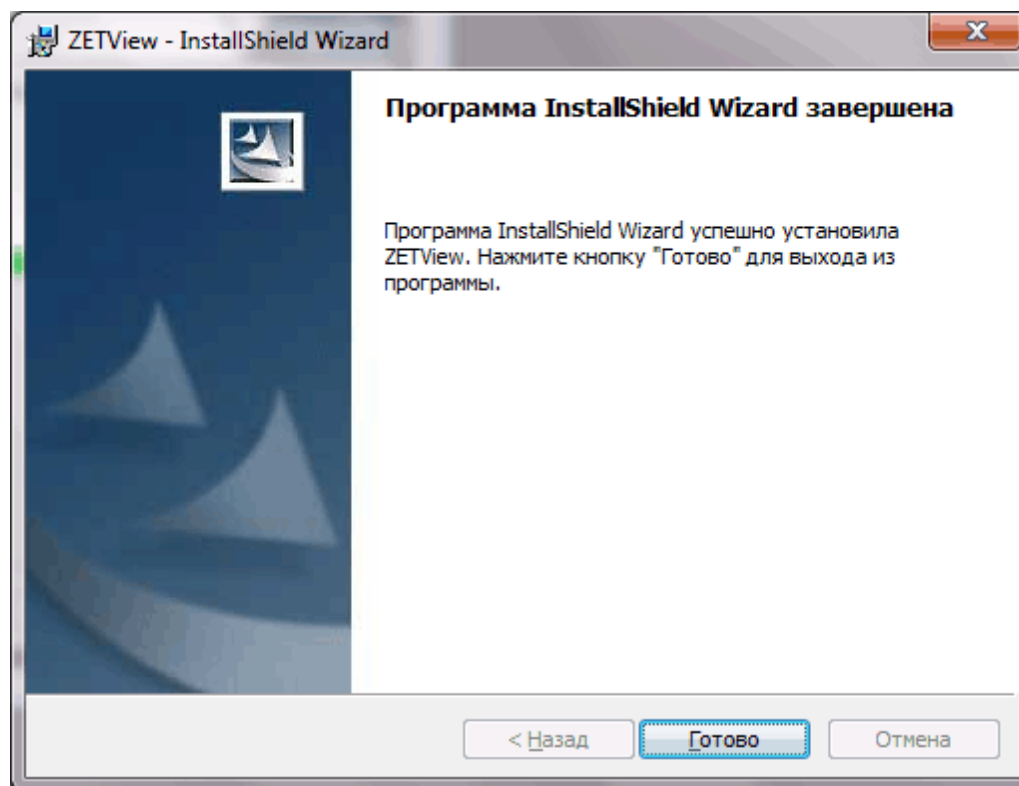


Рисунок 1.3.7. Завершение установки.

## Глава 3. Обновление ZETView

Обновление ZETView осуществляется путем удаления текущей версии и установки новой. Удаление ZETView возможно несколькими способами:

- с помощью установочного диска, поставляемого с приобретаемым ZET-устройством;
- запуском установочного файла «ZETView.msi»;
- через панель управления.

Для удаления ZETView с помощью установочного диска необходимо вставить диск в дисковод компьютера. При этом запустится программа установки программного обеспечения ZETLab и ZETView. В главном окне программы установки программного обеспечения ZETLab и ZETView (рисунок 1.4.1) необходимо выбрать пункт «Установка программного обеспечения», при этом программа предложит установить ZETLab или ZETView (рисунок 1.4.2) – для удаления ZETView необходимо выбрать «Установка ZETView». При этом запустится программа изменения/исправления/удаления ZETView (рисунок 1.4.3).

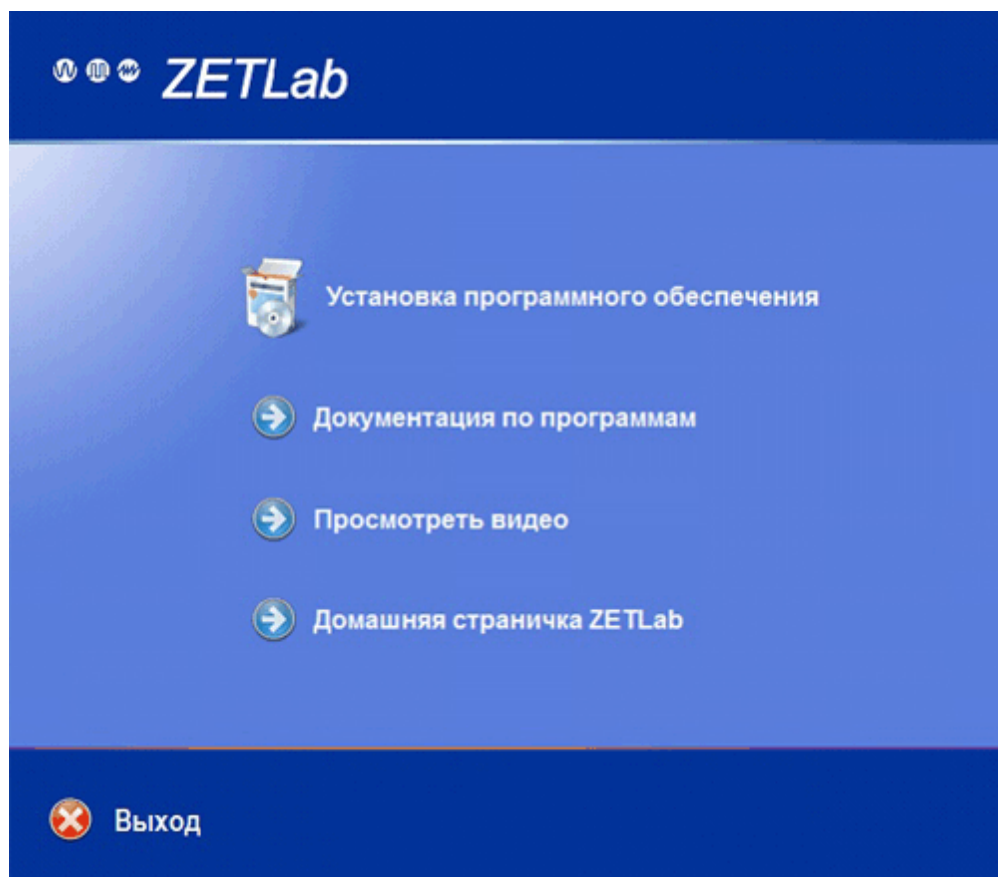


Рисунок 1.4.1

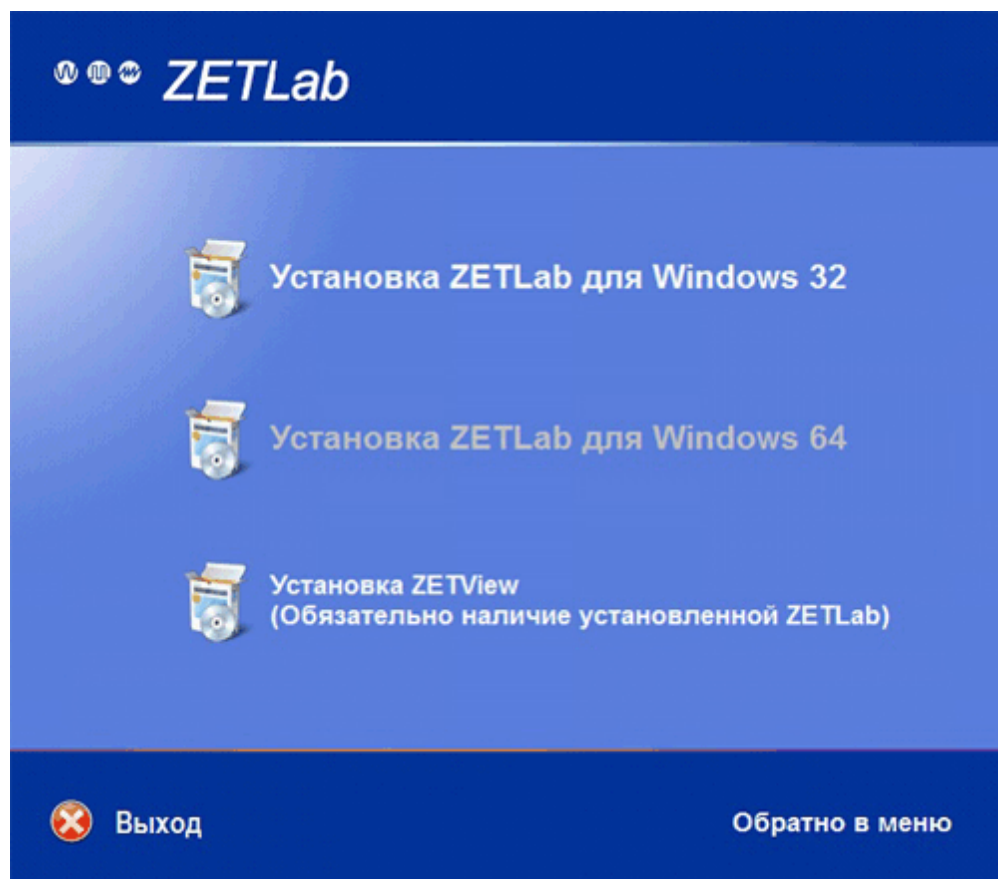


Рисунок 1.4.2

Для удаления ZETView с помощью установочного файла «ZETView.msi» необходимо запустить файл двойным кликом «мыши». При этом запустится программа изменения/исправления/удаления ZETView, для выбора последующих действий необходимо нажать кнопку "Далее", после чего программа предложит выбрать необходимое действие: «Изменить», «Исправить», «Удалить» (рисунок 1.4.4). Для удаления ZETView необходимо выбрать пункт «Удалить».(рисунок 1.4.3).



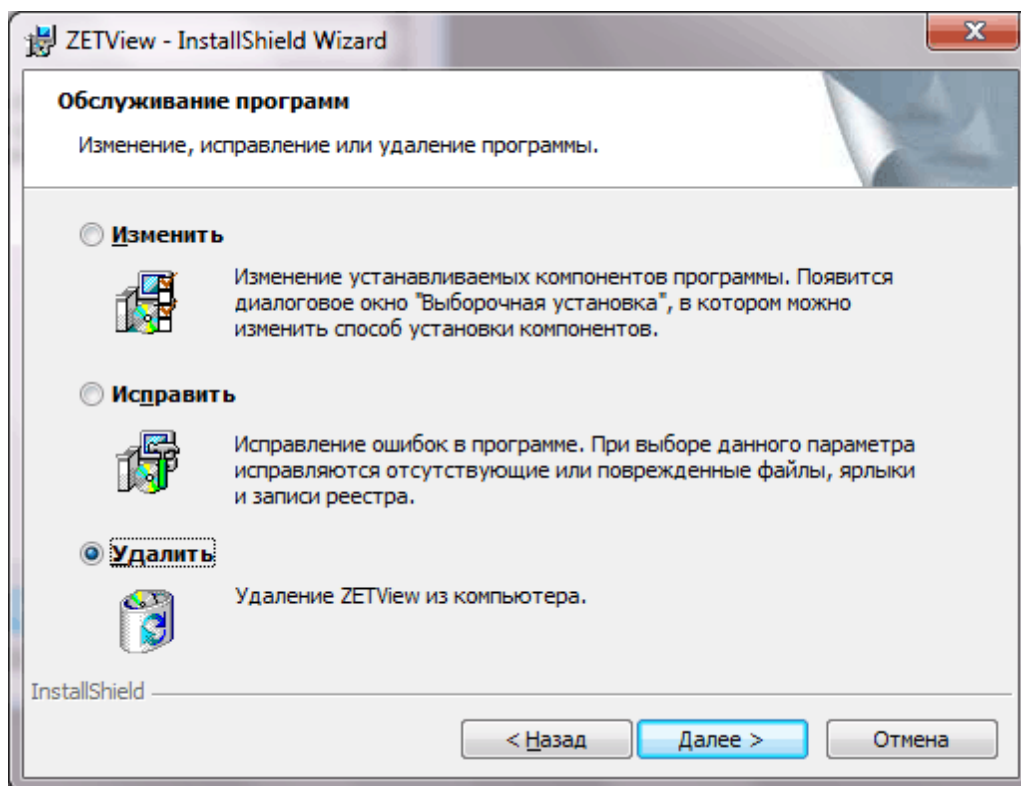


Рисунок 1.4.3. Удаление ZETView

Далее программа запросит подтверждение на удаление. Для подтверждения удаления необходимо нажать кнопку «Удалить» (рисунок 1.4.4).

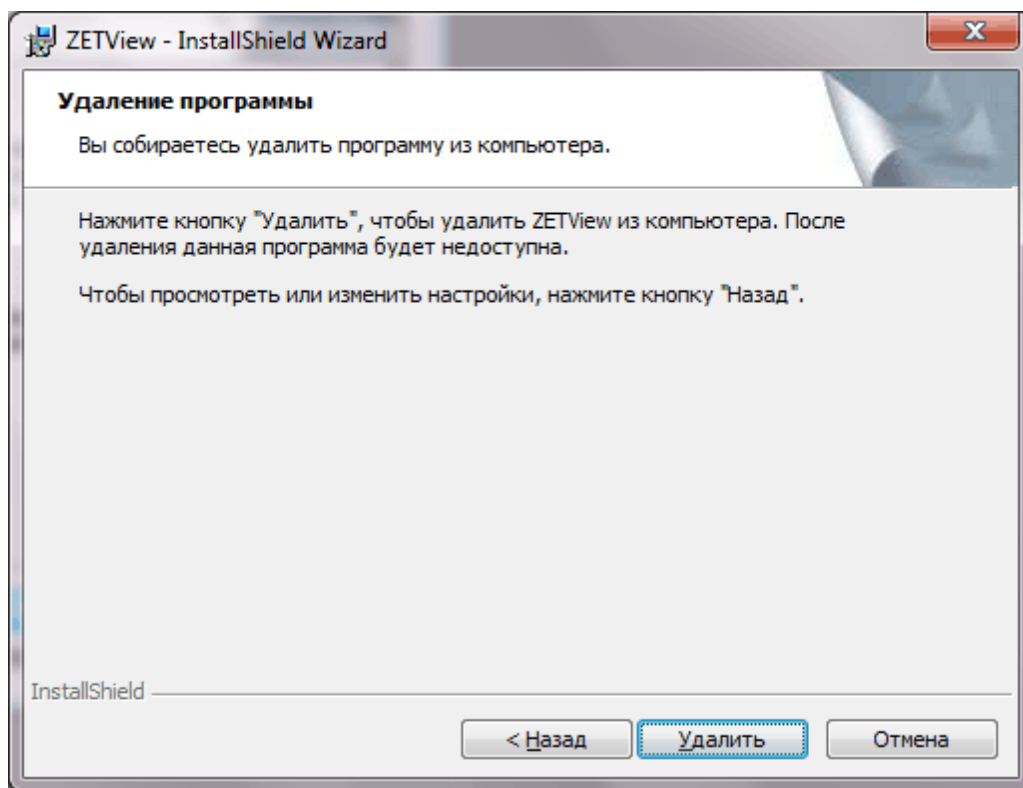


Рисунок 1.4.4.Подтверждение удаления

После подтверждения, запустится процесс удаления (Рисунок 1.4.5).

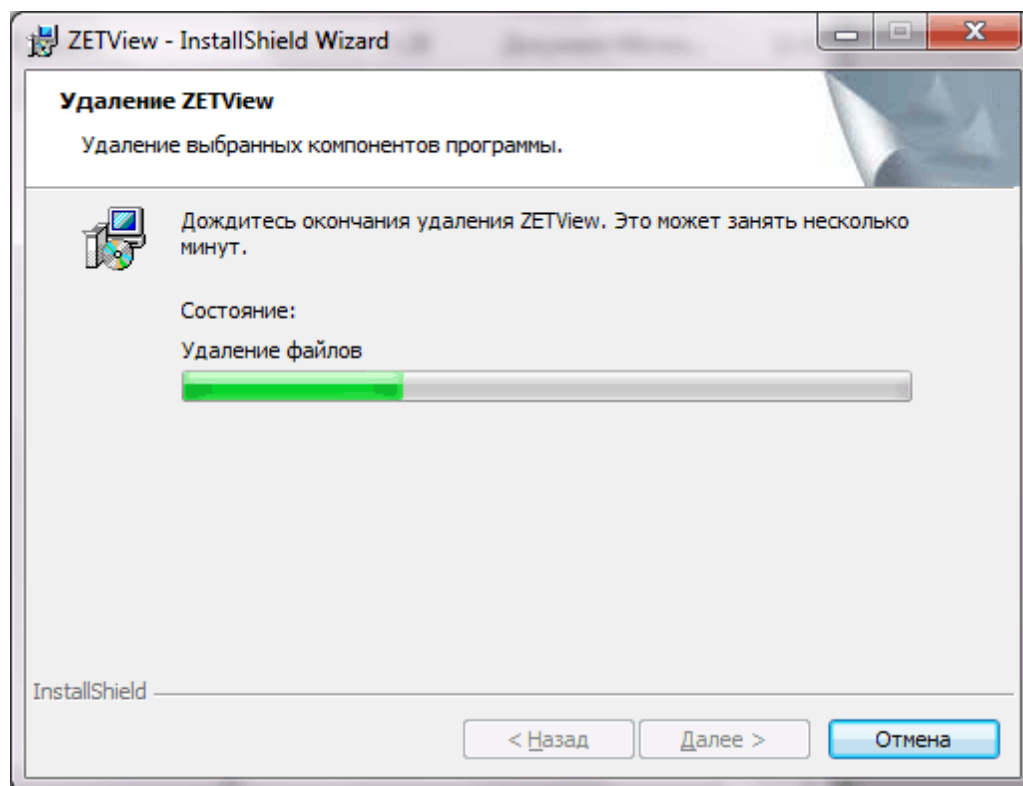


Рисунок 1.4.5. Процесс удаления ZETView

По окончании процесса удаления, установщик сообщит об успешном удалении программы.

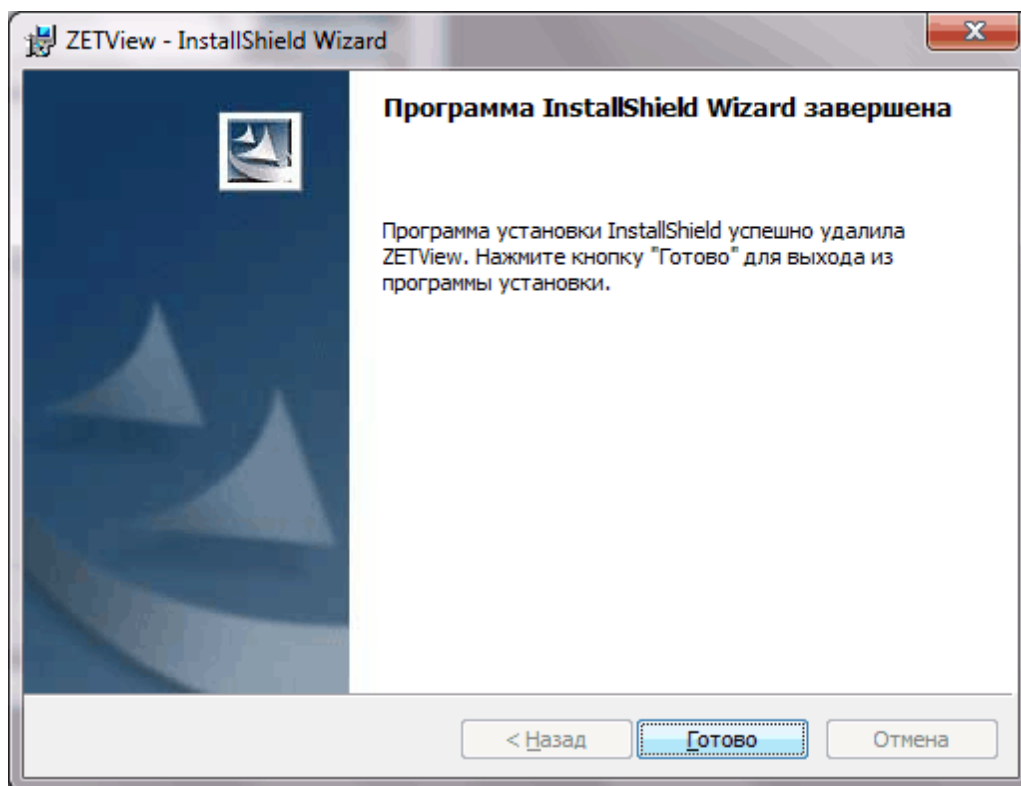


Рисунок 1.4.6. Окончание удаления.

Для удаления ZETView через панель управления необходимо:

- 1) зайти в меню «Пуск» панели ОС Windows и выбрать пункт «Панель управления», при этом отобразится окно «Настройка параметров компьютера» .
- 2) в окне «Настройка параметров компьютера» выбрать пункт «Программы и компоненты», при этом откроется окно «Удаление или изменение программы».

*Примечание: при отображении содержимого окна «Настройка параметров компьютера» в режиме категорий (выбор режима отображений осуществляется в правом верхнем углу окна из списка «Просмотр») выбрать пункт «Удаление программ».*

- 3) в окне «Удаление или изменение программы» выбрать «ZETView» и нажать кнопку «Удалить», при этом будет запрошено подтверждение удаления (рисунок 1.4.5), для удаления ZETView необходимо нажать кнопку «Да»,

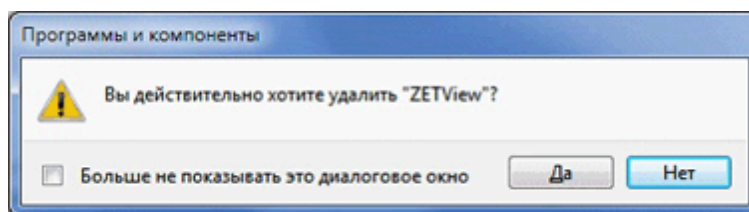


Рисунок 1.4.7

- 4) после подтверждения удаления начнется процесс удаления ZETView, на экране отобразится окно с индикатором процесса удаления и оценкой оставшегося времени (рисунок 1.4.6),

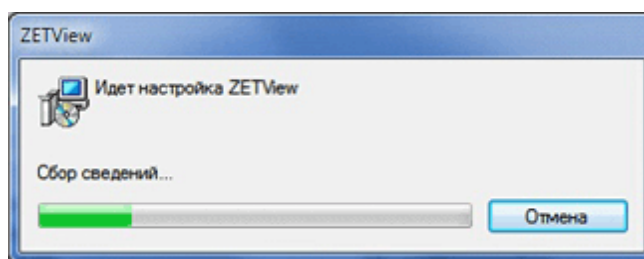


Рисунок 1.4.8

- 5) по окончании удаления операционная система предложит перезагрузить компьютер (рисунок 1.4.7). Перед последующей установкой ZETView рекомендуется перезагрузить компьютер.

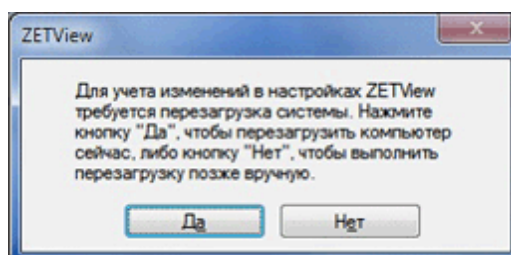


Рисунок 1.4.9

Часто, с выходом новой версии ZETView, также обновляется ZETLab. В этом случае нужно сначала удалить ZETView, как описано выше, затем аналогичным образом удалить ZETLab. Далее установить новую версию ZETLab и потом новую версию ZETView.

## Глава 4. Запуск ZETView

SCADA система ZETView представляет собой многооконное Windows

приложение, в котором разработчик АСУ ТП производит манипуляции по его созданию.

Запустить SCADA ZETView можно используя значок на рабочем столе или через главное меню( Главное меню->ZETLab->ZETView). Также можно запустить ZETView через ZET-панель(ZETLab): кликните на панели кнопку «Автоматизация», далее выберите в меню пункт «Scada ZETView». Если вы не видите ZET-панели, убедитесь что, она не свернута и запущена. Запустить ZET-панель можно используя значок на рабочем столе или через главное меню ( Главное меню->ZETLab->ZETPanel).

В левой верхней части окна ZETView располагается главное меню программы ZETView. В верхней части - лента (панели инструментов) и панель быстрого доступа. Центральную часть окна ZETView занимают собственно страницы проекта. Слева располагается окно компонентов, справа - окно свойств, а в нижней части окна - окно отладочных сообщений и другой служебной информации.

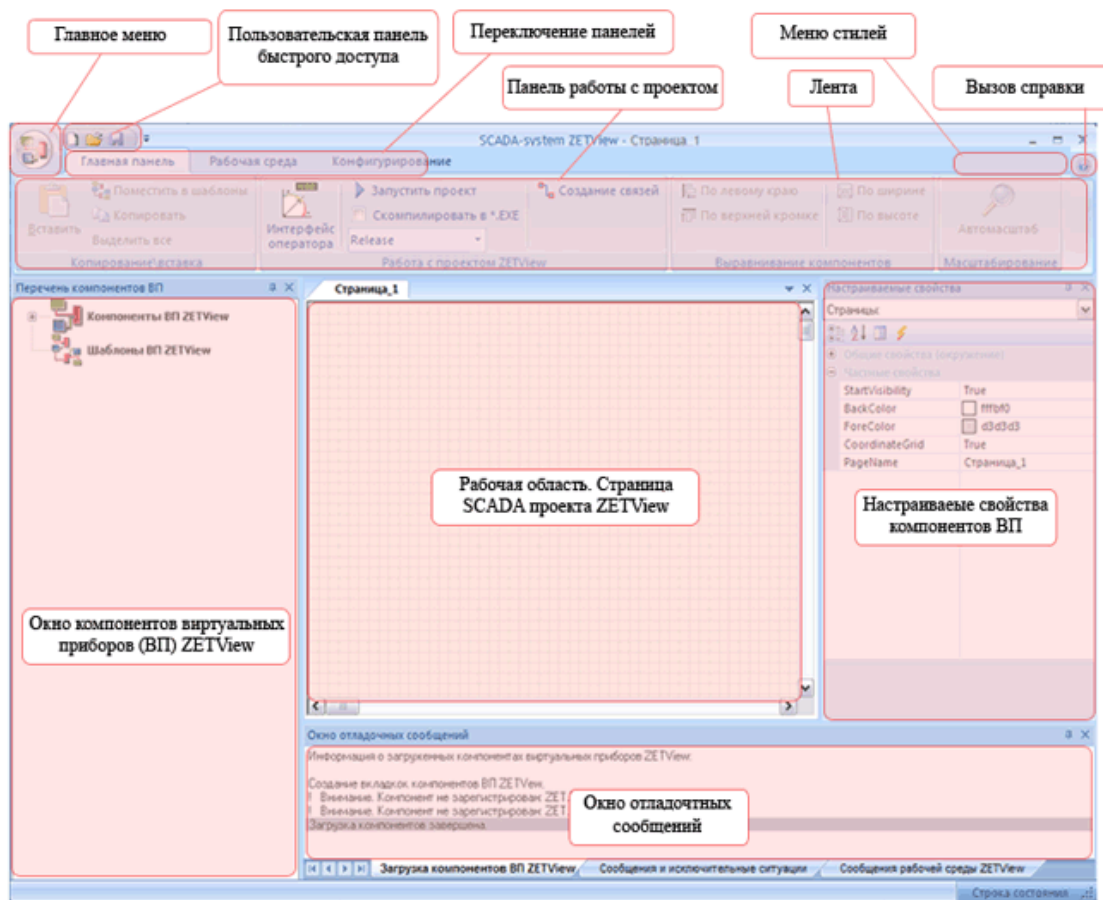


Рисунок 1.5.1. Вид окна ZETView

Компонент – часть проектируемой системы, выполняющая возложенное на нее

законченное действие (например, «Кнопка» реагирует определенным образом на воздействие оператора, а именно, при нажатии на нее можно отключить, либо включить подсоединенный к ней виртуальный прибор).

## Главное меню

Предназначено для работы с проектом в целом.

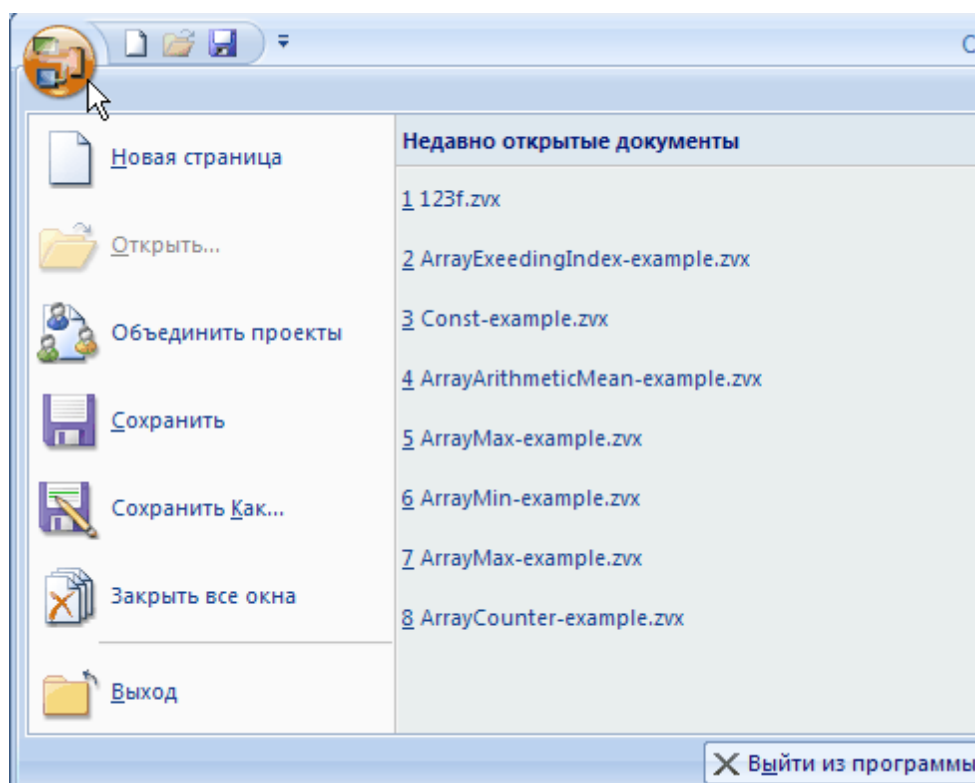


Рисунок 1.5.2. Главное меню

- «Новая страница» - добавление страницы в проект.
- «Открыть» - выбор существующего проекта для просмотра или редактирования.
- «Объединить проекты» - добавление в текущий проект страницы выбранного проекта. Открыв проект и выбрав этот пункт, в появившемся окне выбирается проект, который необходимо добавить. Страницы добавляются после страницы исходного проекта.
- «Сохранить» - сохранение открытого проекта с тем же именем и в той же директории.
- «Сохранить как» - сохранение нового проекта или копии уже имеющегося проекта с выбором имени и директории.

- «Закреть все окна» - закрытие всех проектов.
- «Выход» - Выключает ZETView.

### **Пользовательская панель быстрого доступа**

Панель быстрого доступа является настраиваемой и содержит набор команд, независимых от отображаемой в данный момент вкладки основной панели работы с проектом (ленты). Позволяет получить доступ к основным используемым командам при свернутой ленте, экономя, тем самым, рабочее пространство окна ZETView. На панель быстрого доступа можно добавлять кнопки, выполняющие команды, и перемещать панель в одно из двух возможных местоположений (настройка панели быстрого доступа – см. главу 6)

### **Лента**

Область отображения панелей работы с проектом:

- Главная панель
- Рабочая среда
- Конфигурирование

«Главная панель» является одной из наиболее используемых при работе с ZETView. Позволяет работать с буфером обмена, выбирать режимы работы с проектом, устанавливать одинаковые размеры для выделенных компонентов и выравнивать их расположение на странице, настраивать окно программы ZETView. Настройка интерфейса окна ZETView осуществляется на панели «Рабочая среда». Контроль конфигурации ZET-устройств осуществляется с помощью команд, расположенных на панели «Конфигурирование».

### **Меню стилей**

Позволяет выбирать вид окна ZETView.

### **Вызов справки**

Кнопка. При нажатии открывает файл справки.

### **Окно компонентов виртуальных приборов (ВП) ZETView**

Окно компонентов содержит перечень компонентов виртуальных приборов ZETView. Компоненты в окне могут располагаться по группам (например,



предназначенные для измерения параметров или осуществления логических операций), в алфавитном порядке, по типу и т.д. Добавить компонент в проект можно не только из окна компонентов, но и из контекстного меню страницы.

### **Рабочая область**

Область для размещения компонентов и создания связей.

### **Окно свойств**

Окно свойств предназначено для просмотра и изменения свойств компонентов и страниц проекта. Свойства в окне разделены на группы: общие (свойства, которыми обладают все компоненты) и частные (индивидуальные настройки компонента). Отдельно выделены параметры, отвечающие за линейные размеры и расположение компонентов на странице проекта.

### **Окно отладочных сообщений**

В окне отладочных сообщений отображается информация о загрузке компонентов в библиотеку SCADA системы ZETView, сообщения об ошибках и исключительных ситуациях, а также сообщения рабочей среды ZETView.

## **Глава 5. Настройка ZETView**

Одно из очевидных преимуществ SCADA системы ZETView заключается в возможности настроить окно программы по своему вкусу. Новичку удобнее держать все окна открытыми, чтобы быстрее находить нужные компоненты в окне устройств и видеть настройки активного компонента. Но любая функция в ZETView может быть вызвана с помощью контекстного меню, либо нажатием горячих клавиш.

### **Настройка панели быстрого доступа**

Панель быстрого доступа позволяет получить доступ к часто используемым функциям без затрат времени на их поиск и переключение вкладок ленты. Состав кнопок панели полностью настраивается пользователем. Чтобы войти в меню настройки панели быстрого доступа необходимо нажать на стрелку справа от панели.

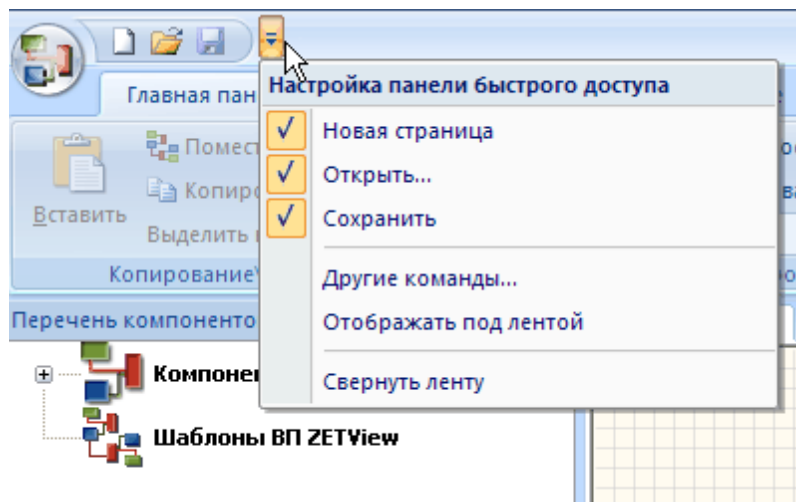


Рисунок 1.6.1. Меню настройки панели быстрого доступа

Добавление/удаление кнопок работы с проектом «Новая страница», «Открыть» и «Сохранить» возможно установкой/снятием соответствующих флагов в меню настройки панели быстрого доступа. Кнопка «Отображать под лентой» переносит панель быстрого доступа под ленту, как показано на рисунке ниже.

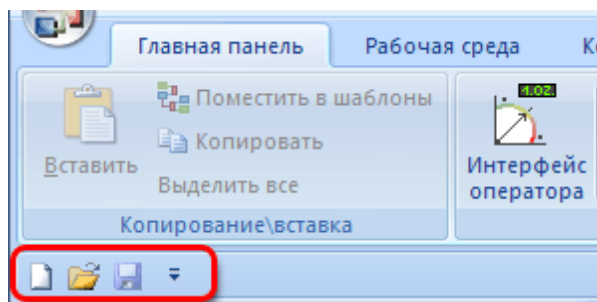


Рисунок 1.6.2. Вид меню быстрого доступа, расположенного под лентой

Флаг «Свернуть ленту» сворачивает/разворачивает ленту. Если лента свернута, отображаются только названия вкладок панелей работы с проектом, а сами панели открываются в всплывающей режиме, т.е. при клике левой кнопки мыши по какой-либо вкладке, соответствующая панель разворачивается, перекрывая собой рабочую область программы. Панель будет отображаться, пока не будет выбрана какая-либо команда, или пока не будет произведен клик кнопкой «мыши» в свободном месте рабочего пространства ZETView.

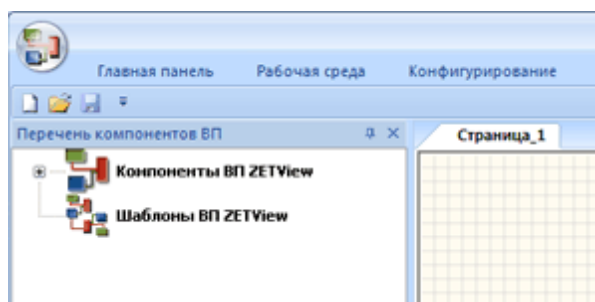


Рисунок 1.6.3. Вид окна ZETView при свернутой ленте

Чтобы восстановить ленту, необходимо снять флаг «Свернуть ленту» в меню настройки панели быстрого доступа. Двойной клик по вкладке «Главная панель», «Рабочая среда» или «Конфигурирование» также возвращает или убирает ленту.

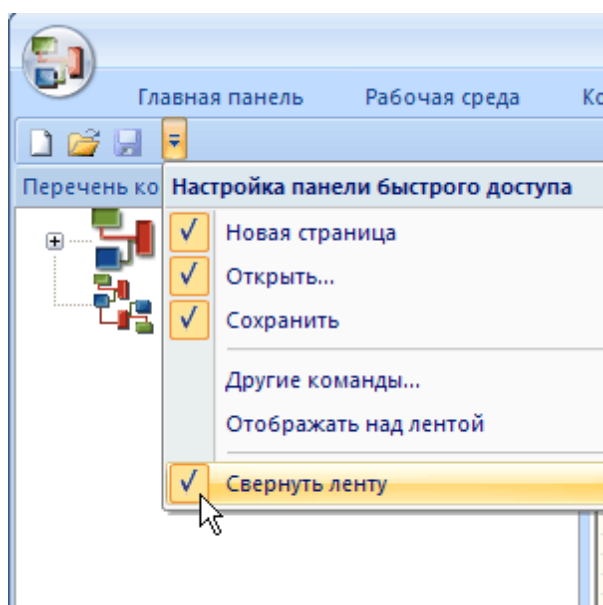


Рисунок 1.6.4. Меню настройки панели быстрого доступа

Функция «Другие команды» вызывает окно «Настройка», которое позволяет поместить дополнительные кнопки на панель.

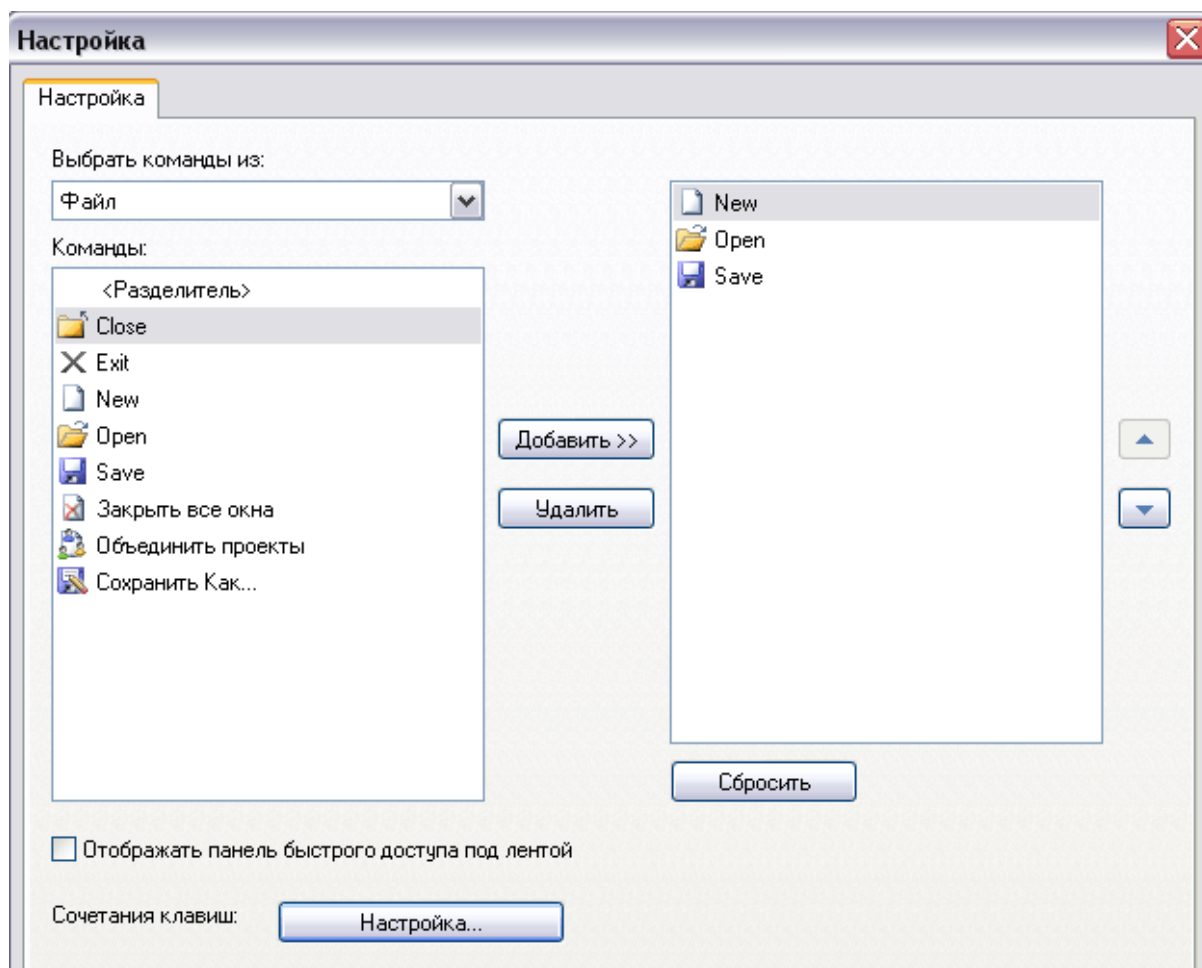


Рисунок 1.6.5. Окно настройки дополнительных кнопок на панели быстрого доступа

В ниспадающем меню «Выбрать команды из» выбирается меню или панель, из которой будут выбраны команды. В зависимости от выбора меняется набор команд в области «Команды». Чтобы поместить кнопку на панель быстрого запуска, выделите щелчком левой кнопкой мыши нужную команду и нажмите кнопку «Добавить». Выбранная команда появится в окне справа. Также добавить команду на панель можно двойным кликом по названию команды в области «Команды». Можно менять порядок выбранных команд с помощью стрелочек, или отменить все выбранные команды, нажав кнопку «Сброс».

### Настройка горячих клавиш

См. рисунок 1.6.6 Перейдите в меню настройки быстрого доступа => другие команды. Нажмите кнопку «Настройка»

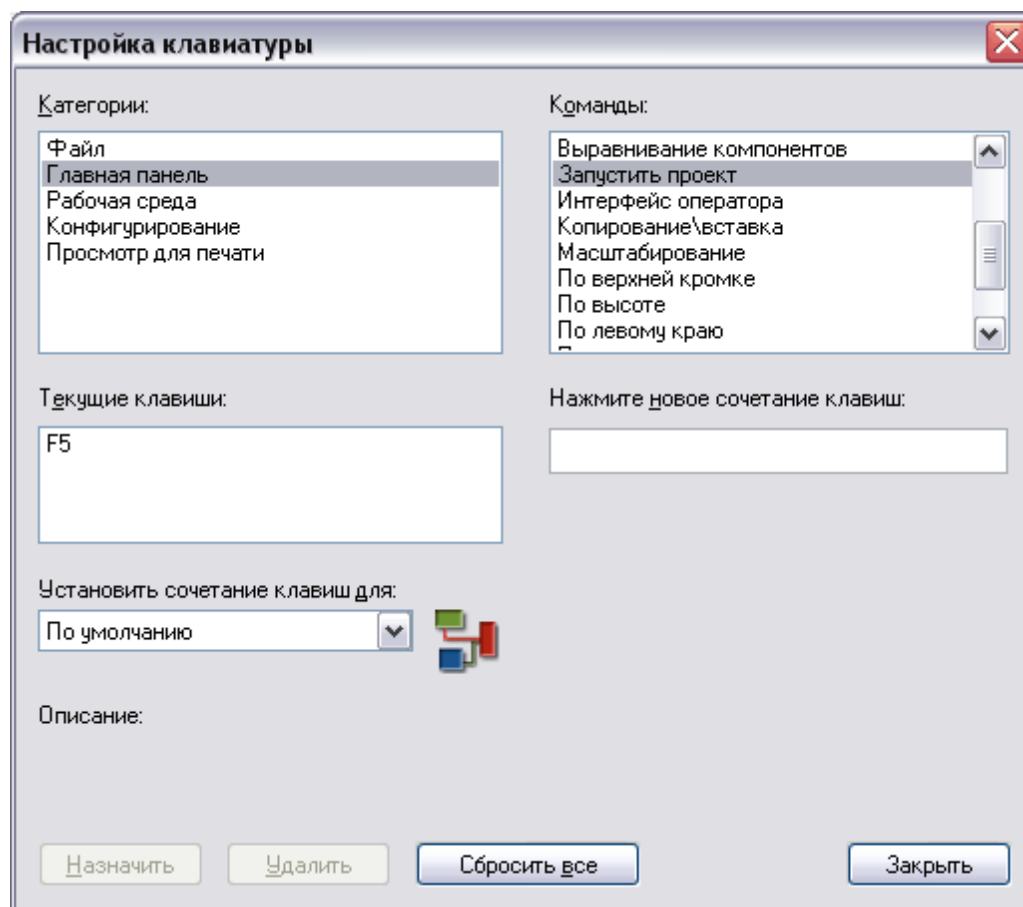


Рисунок 1.6.6. Окно настройки горячих клавиш

Для установки горячих клавиш необходимо выбрать категорию и команду. Если на команду уже установлена комбинация горячих клавиш, она отобразится в окне «Текущие клавиши». Для назначения нового сочетания клавиш необходимо установить курсор «мыши» в поле «Нажмите новое сочетание клавиш» и нажать нужное сочетание, после чего нажать кнопку «Назначить», при этом новое сочетание появится в списке «Текущие клавиши». Чтобы удалить сочетание клавиш, необходимо выделить его в поле «Текущие клавиши» нажать кнопку «Удалить». Кнопка «Сбросить все» сбрасывает все пользовательские настройки горячих клавиш и устанавливает комбинации горячих клавиш по умолчанию.

### Настройка внешнего вида окна ZETView

Внешний вид окна ZETView изменяется в меню стилей (см. рисунок 1.5.1.).

## Настройка рабочей среды

Для настройки рабочей среды перейдите во вкладку «Рабочая среда» над лентой панелей.

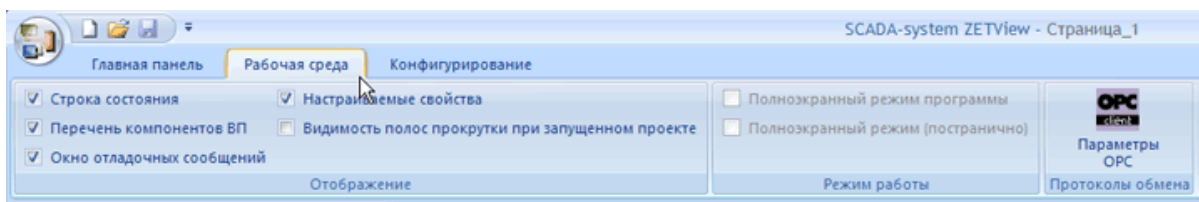


Рисунок 1.6.7. Панель настройки рабочей среды

В разделе «Отображение» можно установить флажки напротив нужных панелей или окон (см. также рисунок 1.5.1.)

## Глава 6. Работа с компонентами

SCADA система ZETView - это среда для графического программирования, другими словами - инструмент проектирования собственных приложений. Основу SCADA-проектов составляют компоненты.

Компоненты добавляются в проект из окна устройств, также называемом библиотекой компонентов. По функциональному назначению компоненты делятся на множество групп: компоненты измерения и анализа, компоненты отображения и индикации, компоненты выбора и управления, компоненты арифметических и логических операций, компоненты работы с массивами, программируемые компоненты и т.д.

Большинство компонентов выполняют простые/законченные действия: «Цифровой индикатор» отображает численное значение, «Частотомер» измеряет частоту сигнала и т.д. Некоторые компоненты предназначены для управления другими компонентами, например, в компоненте «Сценарий» можно изменить свойства других компонентов или вызвать его функцию. Также SCADA система ZETView имеет "служебные" компоненты: «Межстраничная связь», «Менеджер свойств» и т.д. Компоненты не только позволяют выполнять различные задачи, но и имеют множество настраиваемых свойств, подробнее о которых вы узнаете в главе «Свойства компонентов».

Для облегчения "чтения" проекта каждый компонент имеет уникальную пиктограмму, отражающую его назначение. Также, для создания связей с другими компонентами, каждый компонент оснащен контактами разных типов, которые отличаются по цвету и типу передаваемых данным. Подробнее о контактах и соединениях рассказывается далее, в разделе «Создание связей»



Рисунок 1.7.1. Компонент ZETView

В SCADA ZETView компоненты сгруппированы по назначению. Окно компонентов виртуальных приборов Вы можете увидеть в левой части окна программы SCADA ZETView (если окно не отображается см.гл.6 – Настройка SCADA ZETView):

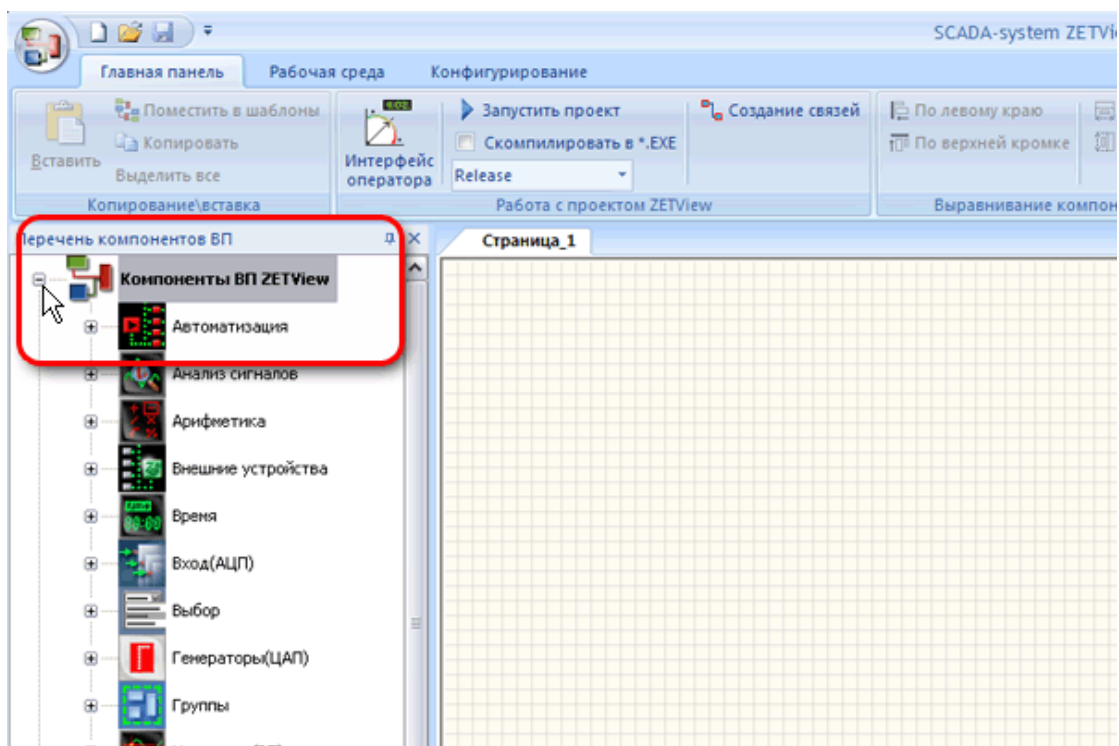


Рисунок 1.7.2. Расположение библиотеки компонентов ZETView

### Как добавить компонент

Открыть список компонентов можно двойным щелчком по нужному пункту или нажав один раз на «+» слева от названия пункта. Для того, чтобы добавить компонент в проект нужно кликнуть на него левой кнопкой «мыши» и, удерживая кнопку «мыши» нажатой, «перетащить» элемент в рабочую область.

Добавить компонент в проект можно через контекстное меню страницы, вызываемое кликом правой кнопки «мыши». При выборе пункта «Компоненты ВП» открывается список разделов библиотеки компонентов, при выборе раздела отображается список компонентов. При щелчке левой кнопки «мыши» по названию компонента выбранный компонент добавляется на страницу проекта в том месте, где было вызвано контекстное меню страницы.



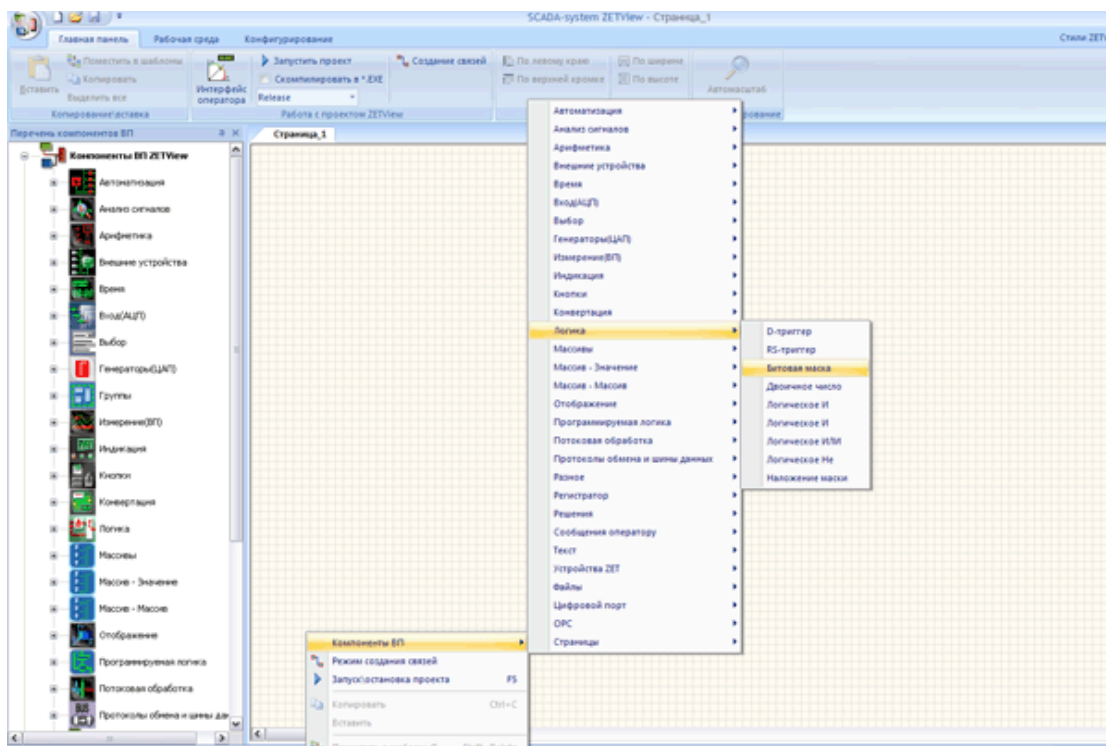


Рисунок 1.7.3. Вызов компонента из контекстного меню

При добавлении компонентов, имеющих отображение и в интерфейсе проектирования и в интерфейсе оператора, в одном интерфейсе, они отображаются в другом интерфейсе в левом верхнем углу страницы

### Как выделить и переместить компонент

Компоненты можно перемещать в рабочей области. Для этого нужно кликнуть по компоненту, тогда вокруг него появится рамка, после этого можно перемещать компонент. Также можно переместить группу компонентов, кликнув левой кнопкой мыши в свободной области и, удерживая кнопку, «растянуть» выделение на нужные компоненты. Вокруг выделенных компонентов также появятся рамки. Пока с компонентов не снято выделение, возможно, их перемещение с помощью "мыши" и стрелок клавиатуры. Чтобы снять выделение необходимо кликнуть на свободное место рабочей области.

### Как скопировать компонент

Скопировать компонент или группу компонентов можно несколькими способами. Выделить компонент или группу компонентов. Затем кликнуть по выделенным компонентам правой кнопкой мыши, после чего появится контекстное меню, в котором выбрать пункт «Копировать». Затем можно кликнуть правой кнопкой

мыши в любом месте рабочей области и в появившемся контекстном меню выбрать пункт «Вставить». Скопированный компонент или группа появятся над компонентами, которые были скопированы со смещением по горизонтали и вертикали на одну клетку. Также вы можете использовать горячие клавиши «Ctrl+C» и «Ctrl+V» и кнопки на главной панели.

### Как удалить компонент

Чтобы удалить компонент или группу компонентов, нужно выделить их и нажать на клавиатуре кнопку Delete.

### Как свернуть компонент

Компонент на странице проекта может быть в «свернутом» или «развернутом» виде (компонент «сворачивается»/«разворачивается» двойным кликом «мыши» по нему). В первом случае он занимает меньше места, во втором - отображаются названия его контактов. Это делает работу в среде ZETView интуитивно-понятной.

Компонент "развернут":

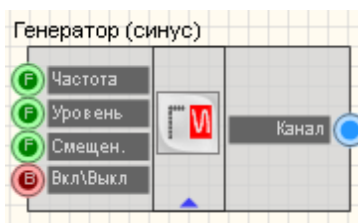


Рисунок 1.5.4 Компонент в развернутом виде

Компонент "свернут":

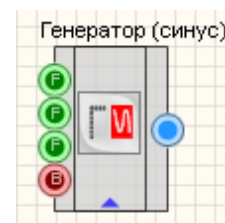


Рисунок 1.5.5. Компонент в свернутом виде

## Глава 7. Соединение компонентов

### Создание связей

Связь между компонентами SCADA осуществляется посредством контактных площадок и соединений. В ZETView контактные площадки различаются по типу передаваемой/получаемой информации, и, в зависимости от типа, имеют различное цветовое и буквенное обозначение. Ниже приведена таблица назначения различных типов контактных площадок.

Вид	Цвет	Обозначение	Интерфейс контакта	Тип передаваемой/ получаемой информации
	Зеленый	F	Число с плавающей запятой	Число с плавающей запятой
	Красный	B	Ключ	Истина/Ложь
	Синий		Id канала	Целое число
	Зеленый	...	Любой тип соединения	Любой
	Черный	T	Текстовая строка	Текст
	Фиолетовый	A	Указатель на числовой массив	Указатель на числовой массив
	Синий	A	Указатель на строковый массив	Указатель на строковый массив
	Салатовый	E	Указатель на книгу/страницу/ячейку Excel	Указатель на книгу/страницу/ячейку Excel

Контакты также классифицируются на 3 типа: входные, выходные и специальный контакт для связи с менеджером свойств. Входные контакты находятся слева, выходные соответственно справа. Контакт для связи с менеджером свойств, позволяющим расширить количество контактов компонента, находится внизу и выглядит как маленький треугольник.

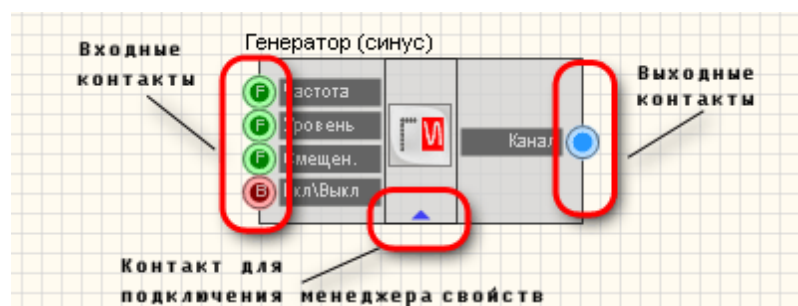


Рисунок 1.8.1. Предназначение контактов

Между контактными площадками двух и более компонентов можно создавать соединения. Соединять можно только контакты одного типа. Исключение составляет лишь контакт для любого типа соединений.

Для создания связи между компонентами необходимо совершить следующую последовательность действий:

1. Поместить нужные компоненты в рабочую область (как это сделать см. в предыдущей главе). Например «Кнопку» (Компоненты ВП ZETView => Кнопки => Кнопка с фиксацией) и «Световой индикатор» (Компоненты ВП ZETView => Индикация => Световой индикатор)
2. Перейти в режим создания связей. Для этого нужно нажать кнопку «Создание связей» (она загорится оранжевым) на главной панели работы с проектом.

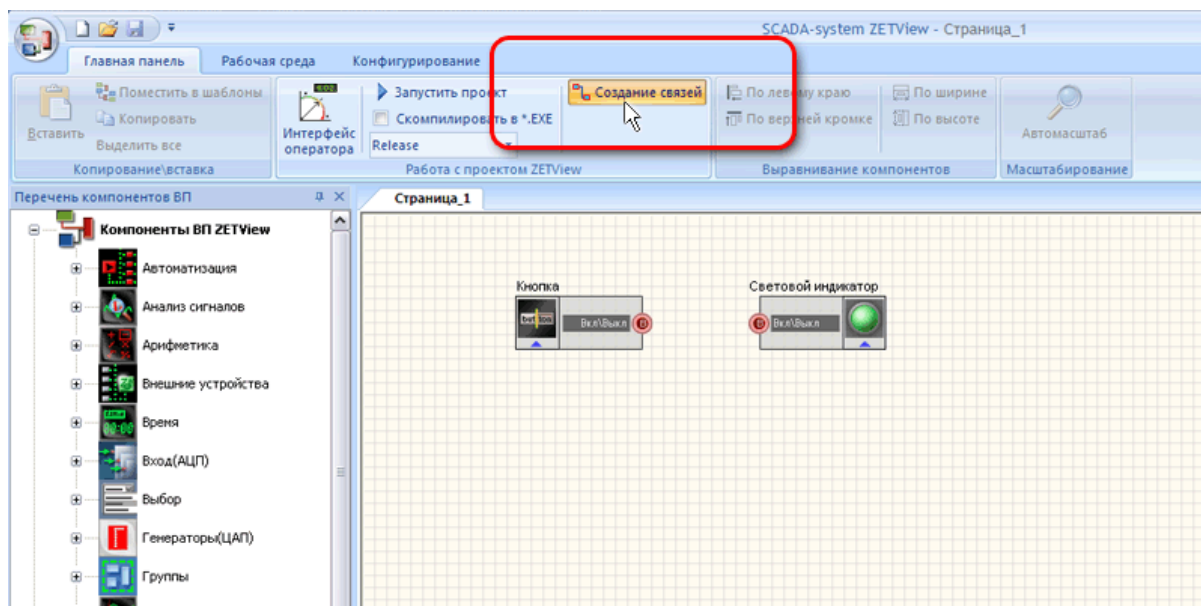


Рисунок 1.8.2. Режим создания связей

3. Чтобы создать соединение необходимо кликнуть по контакту кнопки, при отведении мыши от контакта за ней должна «тянуться» линия, затем кликнуть по контакту светового индикатора.



Рисунок 1.8.3. Процесс создания соединения

*Примечания:*

- Для отмены процесса создания соединения достаточно нажать кнопку клавиатуры <Esc>.
- Связи проводятся от выходного контакта ко входному. Провести связь наоборот нельзя.
- В режиме создания связей при наведении курсора «мыши» на контакты активного компонента рядом с ними появляются всплывающие подсказки о назначении контакта. Компонент становится активным при клике по нему левой кнопкой «мыши», при этом вокруг компонента отображается рамка.

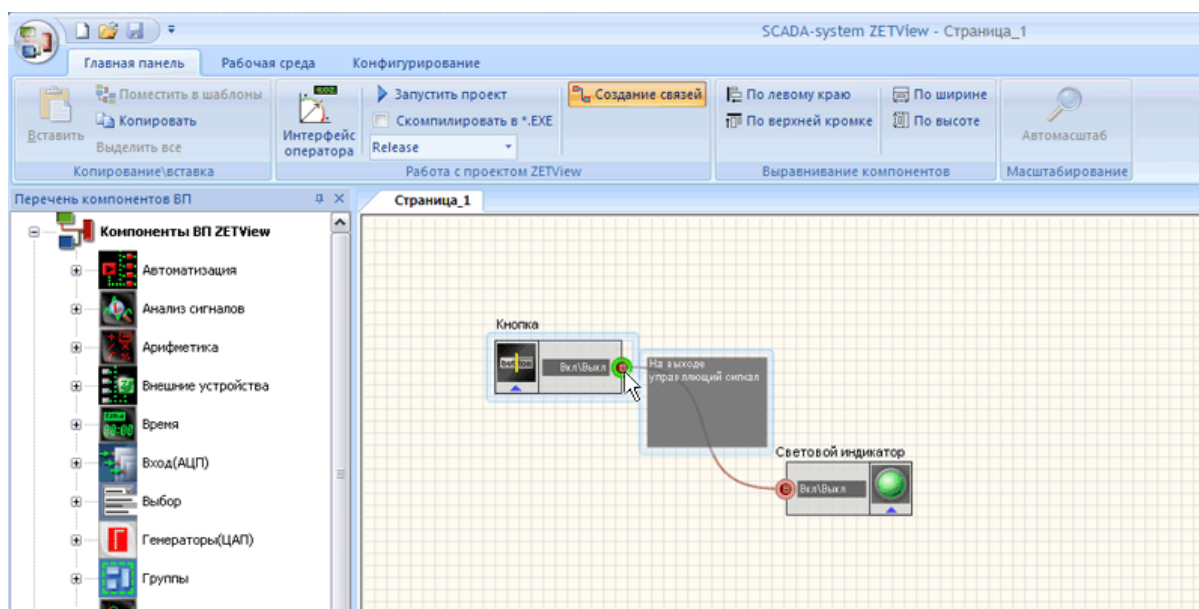


Рисунок 1.8.4. Всплывающие подсказки

### Множественные связи

Большинство контактов могут образовывать множественные соединения. Для создания дополнительного соединения компонента нужно выполнить следующую последовательность действий:

Поместить третий компонент в рабочую область. Для этого нужно, сначала выйти из режима создания связей, нажав кнопку «Создание связей» - она снова должна стать неактивной. Например, можно поместить компонент генератор синусоидального сигнала (Компоненты ВП ZETView => Генераторы (ЦАП) => Синусоидальный сигнал).

*Примечание:*

- При добавлении нового компонента рекомендуется выходить из режима создания связей. Это исключит создание случайных связей и позволит позиционировать элементы в рабочей области.

Далее снова перейдем в режим создания связей. Кликнем левой кнопкой мыши по контакту кнопки и увидим контекстное меню:

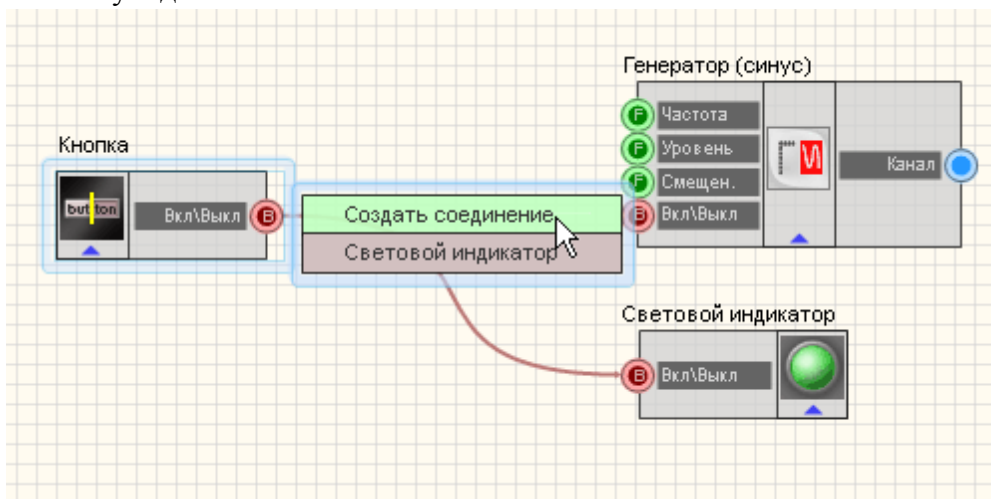


Рисунок 1.8.5. Создание множественных связей

Для образования контактом еще одной линии связи необходимо выбрать пункт «Создать соединение», при этом за «мышью» будет «тянуться» линия связи. Для завершения создания связи необходимо кликнуть по контакту компонента, с которым осуществляется соединение.

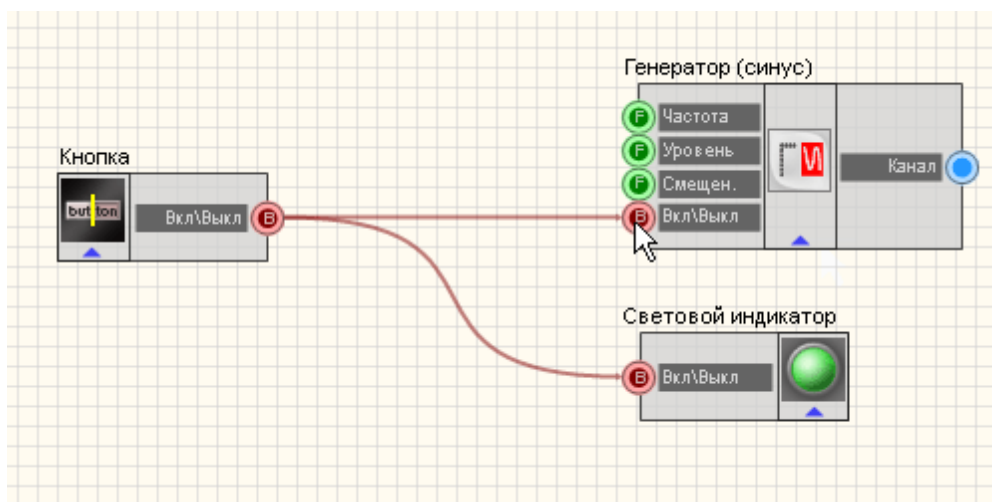


Рисунок 1.8.6. Множественные соединения

### Примечание:

- Создание связей возможно только между контактами одного типа, исключение составляют только контакты «Любой тип соединения». Если между контактами невозможно провести соединение, при попытке образовать связь вокруг контакта появится рамка красного цвета

### Удаление связей

Для удаления соединения необходимо кликнуть левой кнопкой «мыши» по контакту компонента, от которого образовано соединение, и выбрать в появившемся контекстном меню пункт с названием компонента, связь с которым необходимо разорвать:

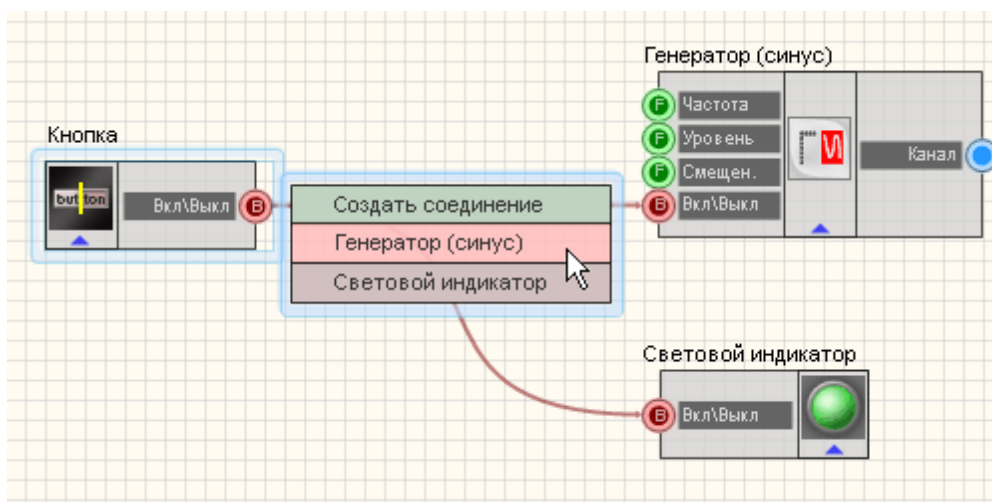


Рисунок 1.8.7. Удаление связи

*Примечание:*

- *Как и при создании соединений, удаляется соединение на элементе, с выходного контакта которого оно создавалось. То есть, если необходимо удалить соединение между «Кнопкой» и «Генератором» (см. рисунок выше), то удалять его нужно на выходном контакте кнопки.*

### **О порядке создания связей**

ZETView обладает особенностью запоминать порядок связей, т.е. если, например, кнопка образует несколько соединений, то при нажатии на кнопку импульс будет передан сначала на тот компонент, связь с которым была проведена первой. В процессе создания проекта важно учитывать, в каком порядке должны передаваться данные подключенным компонентам. Потому что вследствие неправильной последовательности создания связей могут возникнуть трудноуловимые ошибки.

## **Глава 8.Интерфейс оператора и интерфейс проектирования**

Проект ZETView – одна или несколько страниц ZETView с двумя или более программными компонентами, объединенными логическими связями.

Создание проекта включает в себя несколько стадий:

- добавление компонентов в проект
- создание связей между ними
- настройка компонентов
- оформление внешнего вида проекта

Страницы проектов ZETView имеют два интерфейса: оператора и проектирования. Основное отличие заключается в том, что в интерфейсе оператора настраивается внешний вид проекта, а в интерфейсе проектирования отображаются линии связей между компонентами.

При работе со страницей проекта в интерфейсе проектирования, возможно переключение на режим создания связей между компонентами.

Для выбора интерфейса и режима работы служит главная панель инструментов, расположенная в верхней части окна ZETView.



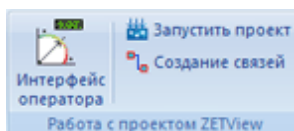


Рисунок 1.9.1. Вид кнопки «Интерфейс оператора» в неактивном состоянии

Для перехода в интерфейс оператора необходимо нажать на пиктограмму «Интерфейс оператора» на главной панели инструментов. При этом внешний вид пиктограммы изменится

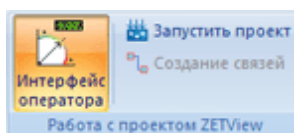


Рисунок 1.9.2. Вид кнопки «Интерфейс оператора» в активном состоянии

*Примечание:*

- В SCADA-системе ZETView есть возможность добавить команды из панелей инструментов на панель быстрого доступа. При добавлении команд "Интерфейс оператора" и "Создание связей" на панель быстрого доступа, можно переключаться между режимами работы и интерфейсами отображения, не обращая к главной панели инструментов ZETView. Внешний вид пиктограмм на панели быстрого доступа будет зависеть от выбранного интерфейса и режима.

На рисунках ниже представлен внешний вид проекта в интерфейсе проектирования и в интерфейсе оператора.

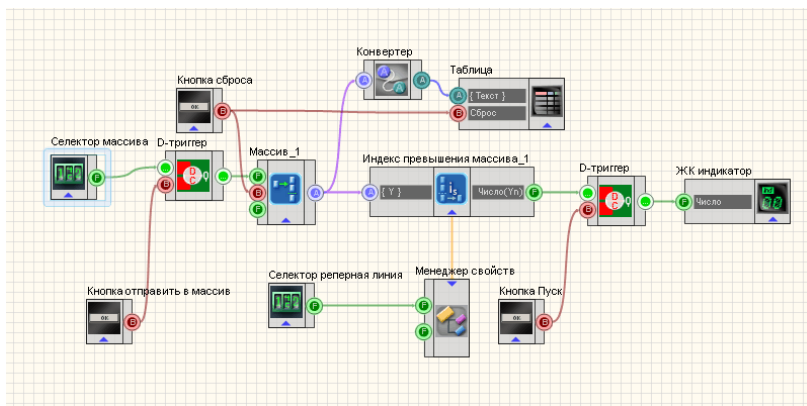


Рисунок 1.9.3. Внешний вид проекта в интерфейсе проектирования.

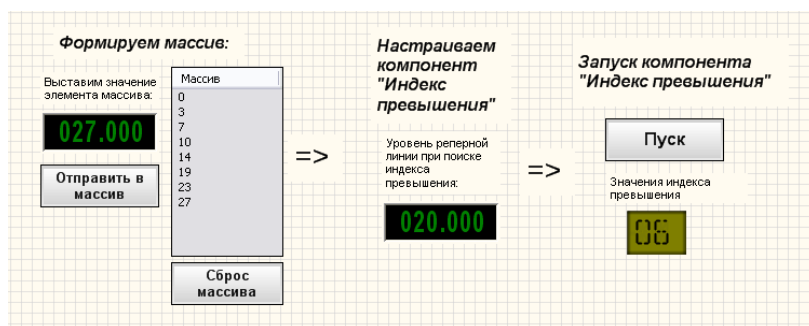


Рисунок 1.9.4. Внешний вид проекта в интерфейсе оператора.

Многие компоненты не отображаются в режиме оператора. Отображаются, как правило, компоненты, управляемые оператором или компоненты выбора, индикации и отображения. В режиме проектирования проектировщик расставляет компоненты, прописывает свойства и создает соединения, в режиме оператора просто настраивается внешний вид компонентов, их расположение, добавляются комментарии.

*Примечание:*

- Для компонентов, имеющих отображение в обоих интерфейсах, можно установить невидимость в одном из них. Для этого следует перейти в тот интерфейс, в котором компонент отображаться НЕ должен, выделить компонент и нажать сочетание клавиш <Ctrl>+<Delete>. Следует отметить, что эта операция не обратима.

Переключение между режимами возможно только на стадии разработки проекта или при редактировании. Когда проект запущен на исполнение, он отображается в соответствии с настройками в режиме оператора. При этом пользователю доступны лишь те компоненты, которые необходимы для управления измерениями и снятия показаний, а все остальные компоненты скрыты, таким образом, во-первых, облегчается работа с проектом, во-вторых, исключается внесение изменений в отлаженный проект.

## Глава 9. Свойства компонентов

Для просмотра свойств компонента, нужно выделить его в рабочей области, тогда свойства отобразятся в окне настраиваемых свойств. Если окно настраиваемых свойств не отображается, его необходимо включить (см. гл.6 Настройка ZETView, пункт

"Настройка рабочей среды"). Также посмотреть свойства компонента можно кликнув правой кнопкой мыши по выделенному компоненту и в появившемся контекстном меню выбрать пункт «Настраиваемые свойства», после этих манипуляций должно появиться окно свойств выделенного компонента.

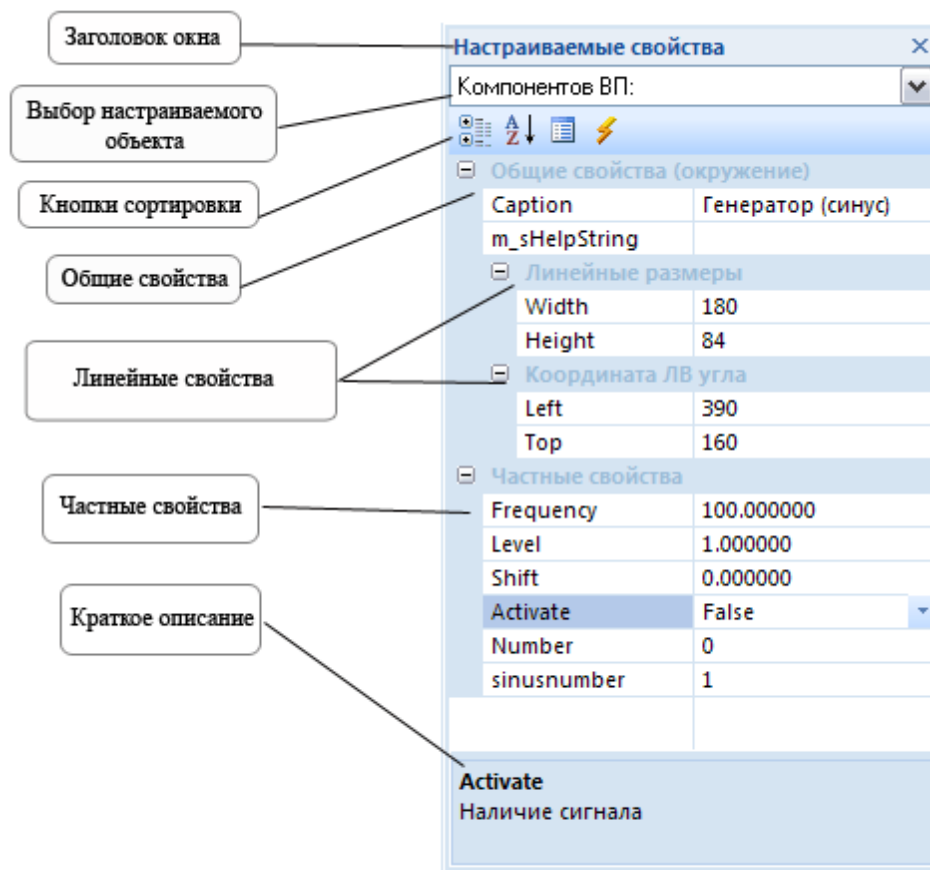


Рисунок 1.10.1. Окно настраиваемых свойств

Свойства компонентов делятся на общие и частные. Общими свойствами обладают все компоненты. Частные свойства индивидуальны для каждого компонента.

Под заголовком окна «Настраиваемые свойства» находится меню выбора настраиваемого объекта. Пункт «Компоненты ВП» выбирается при настройке текущего выделенного компонента, пункт «Страницы» выбирается для настройки вида текущей страницы.

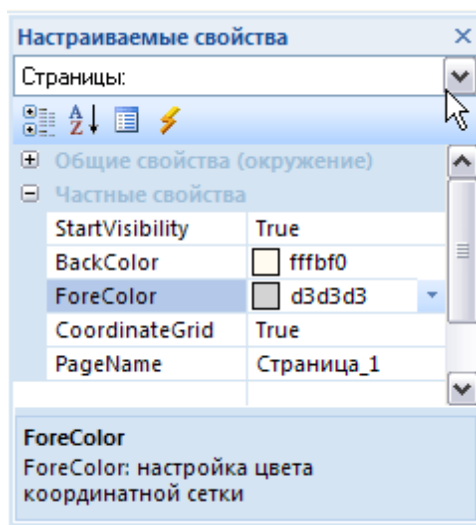


Рисунок 1.10.2. Свойства страницы

В нижней части окна «Настраиваемые свойства», при клике на какое-либо свойство, отображается краткая информация о его назначении.

### Общие свойства компонентов

#### ➤ Общие свойства (окружение):

Caption - каждый компонент в SCADA системе ZETView имеет название (метку). При добавлении компонента в проект он имеет название по-умолчанию, например, "Вольтметр DC" для вольтметра постоянного тока. Проектировщик может изменить название компонента по своему усмотрению, например на "Напряжение на генераторе". Изменить название можно в окне "Настраиваемые свойства" в строке "Caption". При обращении к компонентам в сценарии используется его название, поэтому лучше назначать каждому компоненту индивидуальную метку. Название компонентов отображается над ними только в интерфейсе проектирования, в интерфейсе оператора можно определить название компонента в окне свойств.

m\_sHelpingString - строка идентификатор. По умолчанию отсутствует у всех компонентов, заполняется проектировщиком, если в том есть необходимость, ее заполнение необязательно. Используется в случае, когда свойства одного компонента наследуются другим. Как, например конвертер(числ.масс.-стр.масс) +таблица данных. Также широко используется для написания скриптов ([компонент\\_ "Программный модуль" \[498\]](#))

#### ➤ Линейные размеры:

*Width* - ширина компонента.

*Height* - высота компонента.

- Координаты левого угла:

*Left* - координаты X левого верхнего угла.

*Top* - координаты Y левого верхнего угла.

Линейные размеры и координаты левого верхнего угла задаются для каждого интерфейса отображения

*Примечания:*

- Рекомендуется менять названия компонентов, выставленные по умолчанию, во избежание путаницы в больших проектах.

### Как настраивать свойства

Свойства компонентов меняются в окне настраиваемых свойств. Для изменения какого-либо свойства необходимо поместить курсор в поле значения нужного свойства, удалить текущее значение и ввести новое с клавиатуры, если свойство представлено списком - выбрать подходящее значение из списка.

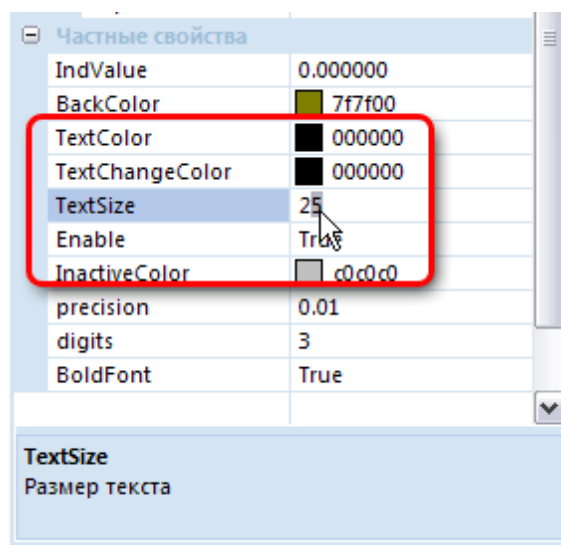


Рисунок 1.10.3. Изменение свойств

Свойства компонентов - это не просто параметры, необходимые для их корректной работы, это основа гибкости SCADA системы ZETView. При различных настройках функциональное назначение одного и того же компонента может быть прямо противоположным. Свойства компонентов могут быть заданы в окне настроек при разработке scada-приложения, или же регулироваться компонентами управления, подключенными к соответствующим контактам компонента (селектор, менеджер свойств).

## Глава 10. Настройка внешнего вида, выравнивание и позиционирование

В ZETView проектировщик сам настраивает внешний вид проекта, таким образом, оператор видит на экране лишь необходимые ему компоненты управления проектом и отображения результатов работы.

### Позиционирование компонентов

Окно "Настраиваемые свойства" в SCADA-системе ZETView содержит разделы "Линейные размеры" и "Координаты ЛВ (левого верхнего) угла", в которых отображаются соответствующие параметры компонента.

*Примечание:*

➤ *Линейные размеры и координаты ЛВ угла – это параметры отображения, а, значит, настраиваются в интерфейсе проектирования и интерфейсе оператора независимо.*

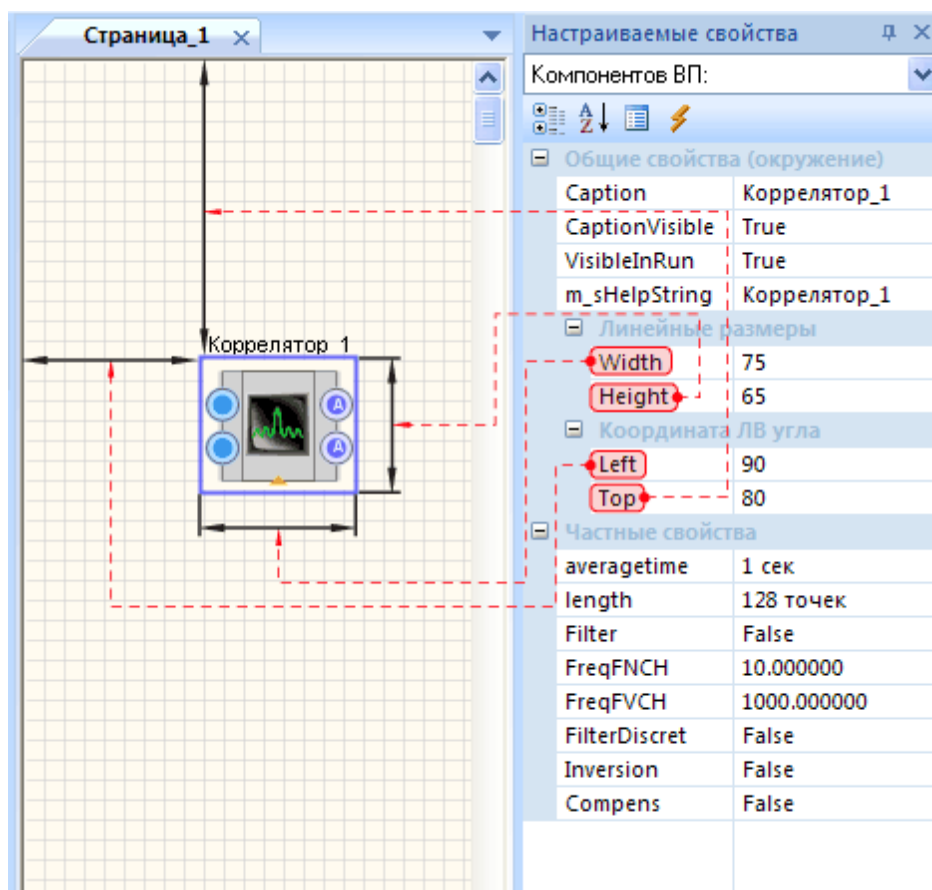


Рисунок 1.11.1. Линейные размеры

Для каждого компонента SCADA системы ZETView устанавливаются линейные размеры (ширина и высота). Единица измерения - пиксели экрана. Ширина компонента задается в строке "Width", а высота - в строке "Height".

Расположение компонента на странице ZETView задается координатами его левого верхнего угла. Координата X задается в строке "Left", а координата Y - в строке "Top". Координаты страницы задаются в пикселях экрана. Точка с координатами 0.0 - левый верхний угол страницы.

Для задания одинаковых размеров нескольким компонентам и выравнивания компонентов относительно друг друга служат соответствующие команды раздела "Выравнивание компонентов" главной панели ZETView:

- по левому краю - выравнивает все выделенные компоненты по левому краю (присваивает свойству Left всех компонентов одно и то же значение),
- по верхней кромке - выравнивает все выделенные компоненты по верхней границе (присваивает свойству Top всех компонентов одно и то же значение),

- по ширине - выравнивает ширину всех выделенных компонентов (присваивает свойству *Width* всех компонентов одно и то же значение),
- по высоте - выравнивает высоту всех выделенных компонентов (присваивает свойству *Height* всех компонентов одно и то же значение).

При выделении нескольких компонентов один из них будет иметь зеленую рамку, остальные - голубую. Выравнивание происходит относительно компонента с зеленой рамкой.

Чтобы не обращаться к окну "Настраиваемые свойства", можно воспользоваться горячими клавишами (см. таблицу ниже). Горячие клавиши меняют размеры и расположение выделенных компонентов. При изменении размеров компонента координаты его верхнего левого угла остаются неизменными.

<Стрелка вправо> сдвигает компонент вправо на 5 единиц	<Shift>+<Стрелка вправо> увеличивает ширину компонента на 5 единиц
<Стрелка влево> сдвигает компонент влево на 5 единиц	<Shift>+<Стрелка влево> уменьшает ширину компонента на 5 единиц
<Стрелка вверх> сдвигает компонент вверх на 5 единиц	<Shift>+<Стрелка вверх> уменьшает высоту компонента на 5 единиц
<Стрелка вниз> сдвигает компонент вниз на 5 единиц	<Shift>+<Стрелка вниз> увеличивает высоту компонента на 5 единиц
<Ctrl>+<Стрелка вправо> сдвигает компонент вправо на 1 единицу	<Ctrl>+<Стрелка вниз> сдвигает компонент вниз на 1 единицу
<Ctrl>+<Стрелка влево> сдвигает компонент влево на 1 единицу	<Ctrl>+<Стрелка вверх> сдвигает компонент вверх на 1 единицу

1 единица = 1 пиксель экрана. Сетка страницы рисуется с шагом 10 единиц.

### Порядок компонентов

Порядок расположения компонентов при их наложении друг на друга устанавливается с помощью контекстного меню для каждого компонента. Контекстное меню вызывается щелчком правой кнопки "мыши" на активном компоненте. В этом



меню необходимо выбрать команду "Порядок", далее "На передний план" или "На задний план". После этого компонент будет расположен на переднем, либо на заднем плане страницы.

### Изменение внешнего вида компонентов в режиме оператора

Внешний вид проекта может быть изменен в соответствии с нормами предприятия. Каждый компонент имеет свойства, позволяющие настраивать его внешний вид. Например, стандартный вид ЖК индикатора можно поменять следующим образом:

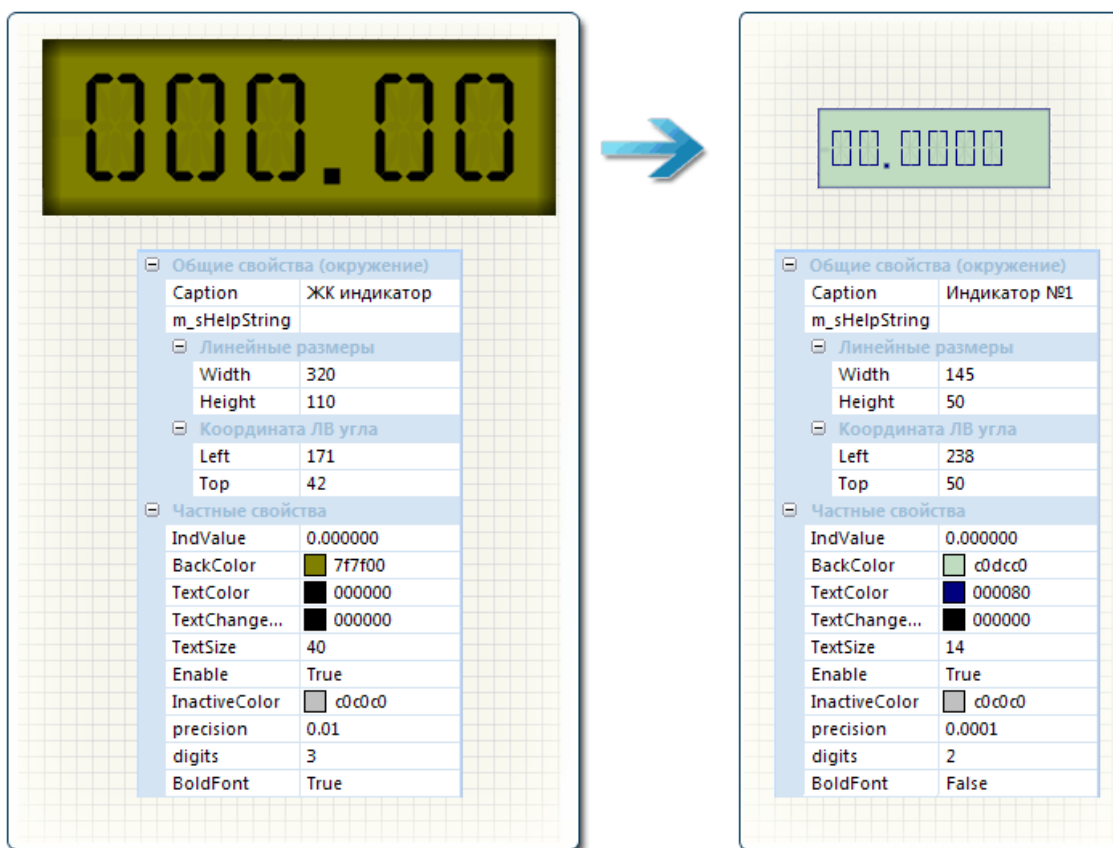


Рисунок 1.11.2. Изменение внешнего вида ЖК индикатора

## Глава 11. Многостраничные проекты

Проект ZETView – часть проектируемой пользователем АСУ ТП. Проект можно создавать как на одной, так и на нескольких страницах, связывая их между собой. Каждая страница, являясь частью проекта, должна выполнять функционально законченное действие, обусловленное задачами пользователя.

Для добавления страницы необходимо перейти в главное меню и выбрать пункт «Новая страница» или использовать комбинацию «горячих клавиш» <Ctrl>+<N>. Вкладка с названием добавленной страницы появится под лентой.



Рисунок 1.12.1. Многостраничный проект

Для того, чтобы удалить страницу из проекта необходимо перейти на страницу, которую нужно удалить, и нажать на крестик справа от вкладок. Если страница не пуста, ZETView запросит подтверждение об удалении.

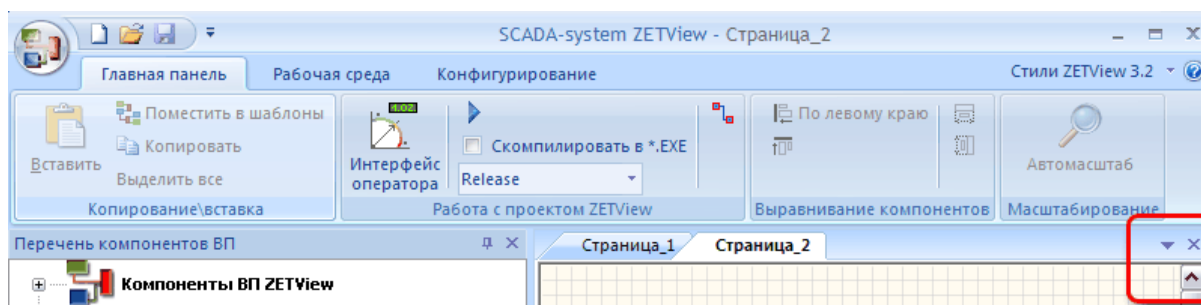


Рисунок 1.12.2. Удаление страницы

Название активной страницы отображается жирным шрифтом, а также на вкладке активной страницы появляется крестик, позволяющий закрыть страницу (т.е. удалить ее из проекта). Кроме того, поле отображения страниц имеет рамку. Цвет рамки соответствует активной на данный момент странице. Названия страниц и другие

свойства страницы настраиваются в окне «Настраиваемые свойства».

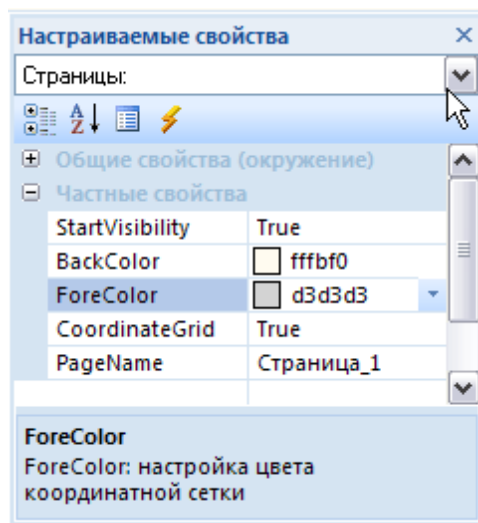


Рисунок 1.12.3. Настраиваемые свойства страницы

Для настройки внешнего вида страницы или изменения имени страницы необходимо перейти на нужную страницу, и в окне «Настраиваемые свойства» в меню выбрать объект «Страница».

StartVisibility (True/False)	Видимость страницы при старте. True - страница будет отображаться при запуске проекта, False - будет скрыта.
BackColor (Цвет из палитры; Цвет фона)	Страница будет иметь заданный цвет не только при разработке, но и при запуске проекта.
CoordinateGrid (True/False)	Видимость координатной сетки. True - координатная сетка отображается, False - не отображается. Примечание: координатная сетка может быть отображена только во время разработки проекта. На странице запущенного проекта координатная сетка не отображается вне зависимости от установки данного параметра.
ForeColor (Цвет из палитры)	Цвет линий координатной сетки.
PageName (Вводится с клавиатуры)	Название страницы. Вводится оператором с клавиатуры. По умолчанию страницы проекта называются "Страница_1", "Страница_2" и т.д. Название страницы может содержать буквы русского и английского алфавита, цифры, и разделительные

СИМВОЛЫ.

В проектах, для создания связей между компонентами, находящимися на разных страницах, используйте специальный компонент «Межстраничная связь» (Компоненты ВП->Разное).

## Глава 12. Запуск, остановка и сохранение проекта. Компиляция в EXE

Когда проект собран и настроен, а связи проведены, остается только сохранить его и протестировать.

Чтобы сохранить проект необходимо нажать кнопку «Сохранить как...» / «Сохранить» в главном меню или нажать сочетание клавиш <Ctrl>+<S>. При сохранении нового проекта ZETView предложит выбрать путь для сохранения файла. Проекты ZETView по умолчанию сохраняются в директории:

C:\Documents and Settings\All Users\ZETLab\SCADA\Projects

При сохранении проекта пользователь может выбрать существующую папку или создать новую. Рекомендуется сохранять проект каждый раз перед его запуском, во избежание потери изменений, при наличии ошибок в проекте.

Для запуска проекта необходимо нажать кнопку «Запустить проект» на главной панели (она загорится оранжевым) или горячую клавишу <F5>. После запуска проекта станут активны элементы управления (кнопки, селекторы, списки) и отображения (графики, таблицы, индикаторы), т.е. они начнут реагировать на манипуляции «мышью» и нажатие клавиш клавиатуры.

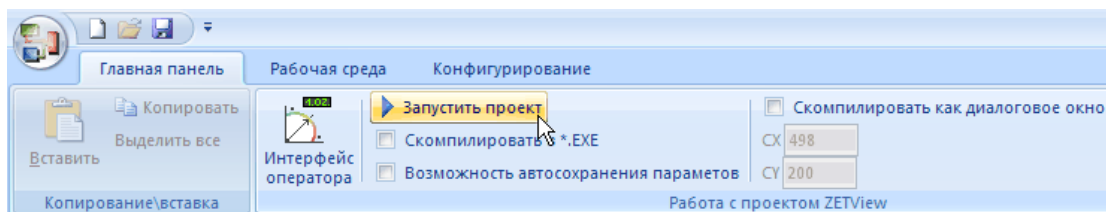


Рисунок 1.13.1. Запуск проекта

Для остановки запущенного проекта необходимо нажать клавишу клавиатуры <F5> или выбрать команду «Запустить проект» на главной панели ZETView.

Сохраненный SCADA-проект имеет расширение \*.zvx. При двойном клике на файл \*.zvx соответствующий проект запускается, но может быть остановлен для редактирования в ZETView. При открытии файла \*.zvx с помощью команды «Открыть» главного меню ZETView проект будет открыт в ZETView и его также можно будет редактировать.

В ZETView предусмотрена возможность компиляции проекта в исполняемое приложение – EXE-файл. Скомпилированный в EXE, SCADA-проект нельзя просмотреть или изменить, его можно только запустить и менять только те параметры, настройка которых предусмотрена проектировщиком.

Для того, чтобы скомпилировать проект в EXE, необходимо установить флаг «Скомпилировать в \*.EXE» на главной панели работы с проектом (при этом должна измениться иконка на кнопке «Запустить проект») и нажать кнопку «Запустить проект».

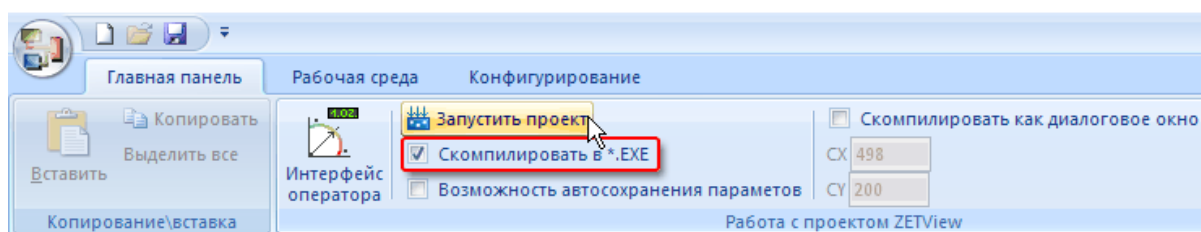


Рисунок 1.13.2.Компиляция проекта в EXE

При компиляции EXE-файла ZETView сообщит, что подготовлен исполняемый модуль программы и предложит его запустить. Исполняемый файл сохраняется в той же директории и имеет то же название, что и исходный \*.zvx файл проекта.

В ZETView также имеется возможность выставить размеры окна при компиляции проекта в \*.EXE. Для этого необходимо установить флаг "Скомпилировать как диалоговое окно" и выставить размеры окна в полях CX и CY. Размеры указаны в пикселях. Можно оценить размеры окна скомпилированного проекта с помощью строки в правом нижнем углу программы, которая показывает текущие размеры окна.

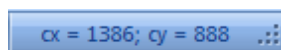


Рисунок 1.13.3. Текущие размеры окна

# Часть 2. Основы проектирования в SCADA ZETView

## Глава 1. Введение

Закончен первый этап изучения ZETView и читатель уже подвел для себя итоги. Если простейшие манипуляции с ZETView и ее компонентами уже не вызывают вопросов и затруднений, то можно смело приступать к изучению второй части руководства. Эта часть носит скорее ознакомительный, нежели обучающий характер. В SCADA ZETView множество компонентов, и их число постоянно увеличивается, человеку, использующему систему впервые, будет трудно сразу разобраться с таким большим количеством компонентов. Поэтому в данной главе будут рассматриваться методы решения ряда микрозадач разной сложности, которые возникнут у любого пользователя в процессе работы со SCADA ZETView, от работы с текстом и простого ввода данных до работы с виртуальными измерительными приборами. Как и другие сложные системы, SCADA ZETView и некоторые ее компоненты обладают своими особенностями работы, о которых также можно узнать из данной главы.

Вторая часть организована так, что необязательно читать ее полностью. Достаточно возвращаться к нужному разделу по мере решения поставленной задачи.

## Глава 2. Ввод данных, индикация и управление

В современном мире редко можно встретить программу, которая бы не информировала пользователя о своих действиях и не отображала промежуточных результатов, тем более в области измерений и обработки информации. Графический интерфейс, элементы управления, различные сообщения и прочие средства взаимодействия с пользователем стали неотъемлемой частью любого приложения. В первой части руководства уже говорилось о возможности компиляции проектов в ехе-файл. Данный формат подразумевает, что оператор не сможет посмотреть, как устроен проект, и прописать настройки в свойствах компонента. Фактически проект будет самостоятельной программой. Поэтому важно обеспечить оператору возможность взаимодействовать с проектом и просматривать результаты.

Компоненты, предназначенные для осуществления ввода данных, находятся в группе «Выбор».

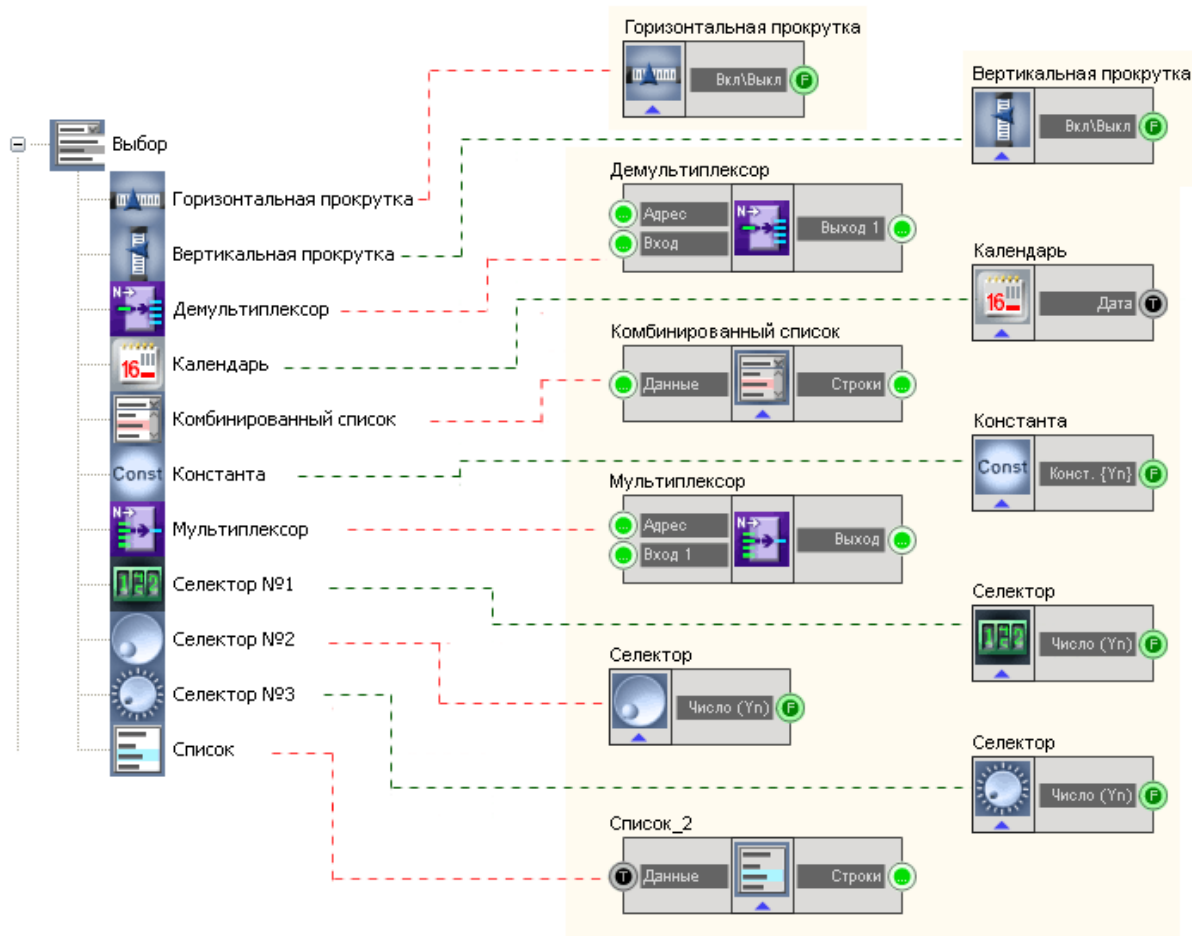


Рисунок 2.2.1. Группа "Выбор".

Для ввода численных значений обычно используются селекторы. В группе «Выбор» доступно пять видов селекторов: Селектор №1 (рис.2.2.2 -б), Селектор №2 (рис.2.2.2 -г), Селектор №3(рис.2.2.2 -в), вертикальный (рис.2.2.2 -д) и горизонтальный (рис.2.2.2 -а) селекторы. Все эти компоненты передают на выход число, выставленное пользователем. Отличаются они лишь внешним видом и, соответственно, удобством использования в том или ином проекте.

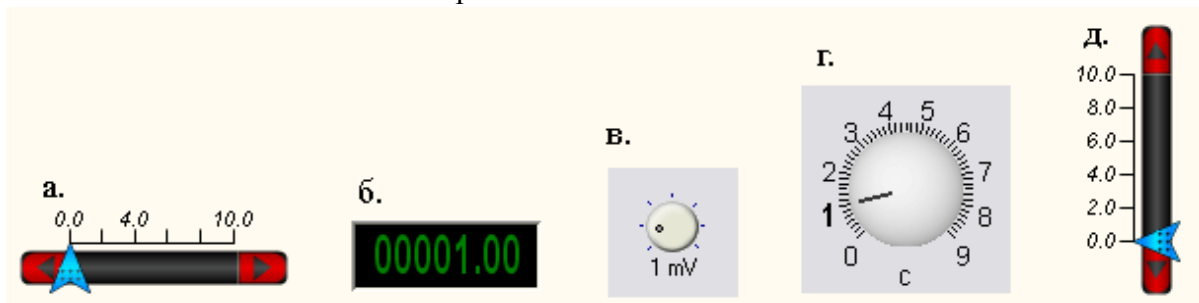


Рисунок 2.2.2. Виды селекторов.

Например, для переключения состояний, удобнее использовать второй селектор, в то время как для ввода точного числа -первый. В селекторе №1 имеется возможность

выставить точное значение (двойной клик мыши по компоненту при запущенном проекте, в появившемся поле ввести нужное число и нажать кнопку "Enter"). В настраиваемых свойствах можно контролировать допустимое вводимое значение и другие параметры.

Логика работы SCADA ZETView рассчитана под обработку сигналов в реальном времени, поэтому важно отличать такие понятия как значение, постоянно передаваемое в канал и значение, передаваемое один раз. Например, значение с селекторов передается один раз. В следующий раз передача осуществляется при изменении значения. Компонент константа также передает указанное значение один раз.

В сложных разветвленных схемах используются компоненты «Мультиплексор» и «Демультимплексор», которые коммутируют необходимый канал по полученному адресу.

Важным элементом выбора являются списки. В ZETView различают комбинированный и обычный списки.

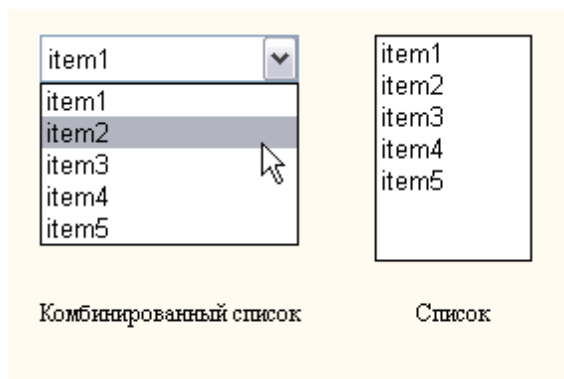


Рисунок 2.2.3. Списки.

Эти компоненты зачастую используются для выбора настроек оператором. Список может быть сформирован вручную или автоматически при подаче на вход строки значений элементов списка.

Для отображения результатов работы в SCADA ZETView предусмотрен целый набор различных индикаторов. Для вывода численных значений используется ЖК индикатор и цифровой индикатор. Зачастую цифровой и ЖК индикаторы используются для отображения показаний вольтметров и других измерительных приборов. Как и многие другие компоненты ZETView, индикаторы рассчитаны для обработки сигналов в режиме реального времени и продолжают отображать последнее полученное значение, пока не получат на вход новое.

Для индикации изменений значений в реальном времени используется индикатор уровня и стрелочный индикатор. Если необходимо сообщить оператору, например, о степени завершенности того или иного процесса, используется индикатор процесса. Как видно из рисунка, в ZETView доступно две разновидности индикатора процесса: принимающий на вход число и управляющий импульс. Первый может отображать уровень завершенности процесса в зависимости от числа, полученного на вход, второй же просто реагирует на полученный импульс и используется для оператора, чтобы тот мог определить, запущен ли процесс, или, вследствие ошибок в настройках или при разработке проекта, запущенный проект «завис». Так же не стоит забывать про световой индикатор, меняющий свое состояние в зависимости от



логического уровня на входе, который фактически является аналогом обычной «лампочки».

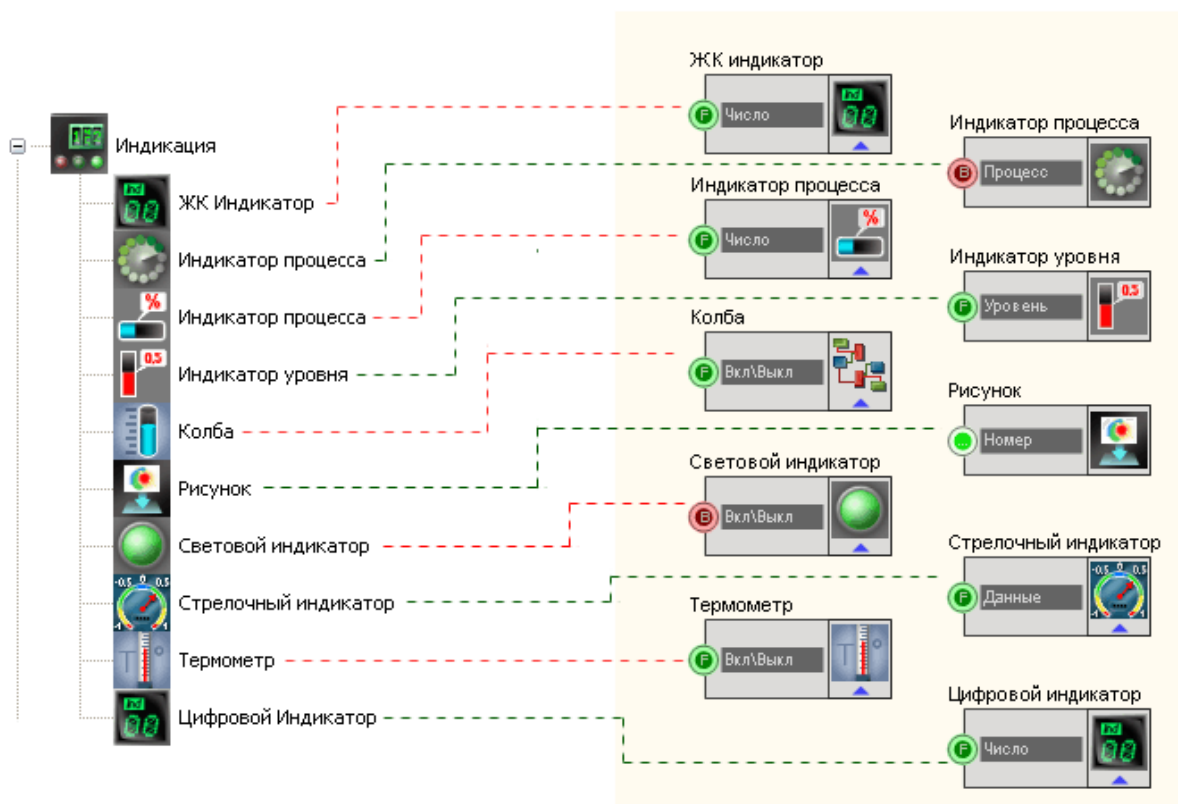


Рисунок 2.2.4. Группа "Индикация".

В управлении многими проектами часто используются кнопки. В SCADA ZETView предлагается несколько видов кнопок. Но фактически они подразделяются на два типа: фиксированного и нефиксированного. Кнопки фиксированного типа при нажатии передают в канал постоянный сигнал определенного логического уровня – «1» или «0». Кнопки нефиксированного типа изначально имеют логический уровень «0», при нажатии посылают единичный импульс «1», после чего снова возвращаются в исходное состояние «0».

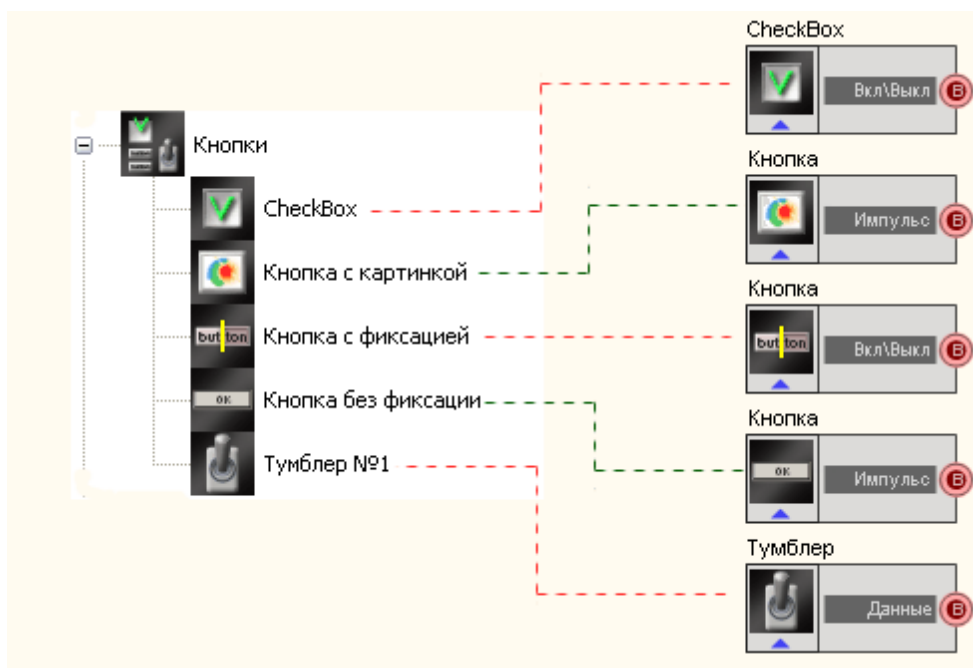


Рисунок 2.2.5. Группа "Кнопки".

На рисунке 2.2.5 изображена группа «Кнопки» в SCADA ZETView. Кроме компонента «Кнопка без фиксации», все кнопки относятся к фиксированному типу, отличаются лишь конфигурацией. CheckBox передает в канал логическую «1», когда установлен флажок, «0» - когда флажок снят. Кнопка с картинкой работает как обычная кнопка с фиксацией, с тем исключением, что вместо кнопки можно выбрать картинку.

Соберем проект SCADA ZETView.

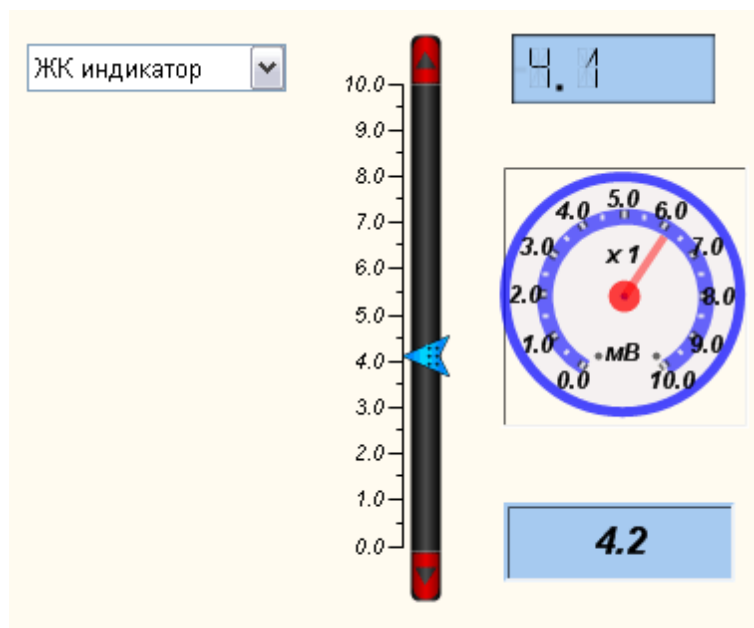


Рисунок 2.2.6. Первый проект.

В этом проекте пользователь из списка выбирает тип индикатора, на который будет выводиться результат с вертикальной прокрутки. На выбранном индикаторе будет отображаться изменение значения при перемещении указателя на вертикальной прокрутке.

Запустим SCADA ZETView и разместим в рабочей области компоненты, указанные ниже, как показано на рисунке 2.2.7.

Группа "Выбор" -> Комбинированный список

Группа "Арифметика" -> Сложение

Группа "Выбор" -> Вертикальная прокрутка

Группа "Выбор" -> Демультимплексор

Группа "Индикация" -> ЖК индикатор

Группа "Индикация" -> Стрелочный индикатор

Группа "Индикация" -> Цифровой индикатор

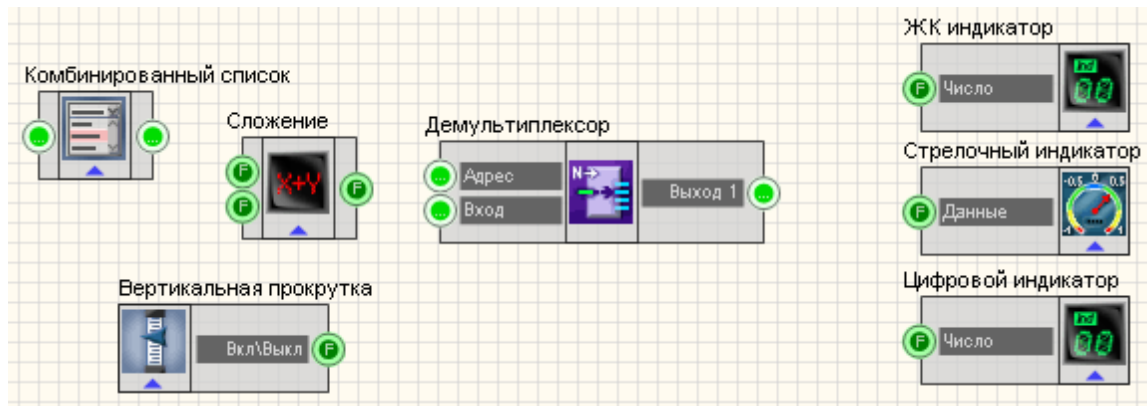


Рисунок 2.2.7. Расставим компоненты.

Компонент Демультимплексор может обладать несколькими выходами, в зависимости от настройки. При получении номера выхода на вход "Адрес" демультимплексор будет передавать на указанный выход данные с контакта "Вход". Настроим демультимплексор. Для этого кликнем правой кнопкой мыши по компоненту и в появившемся меню выберем пункт "Настраиваемые свойства". Тогда в правой части окна SCADA ZETView появится окно настраиваемых свойств.

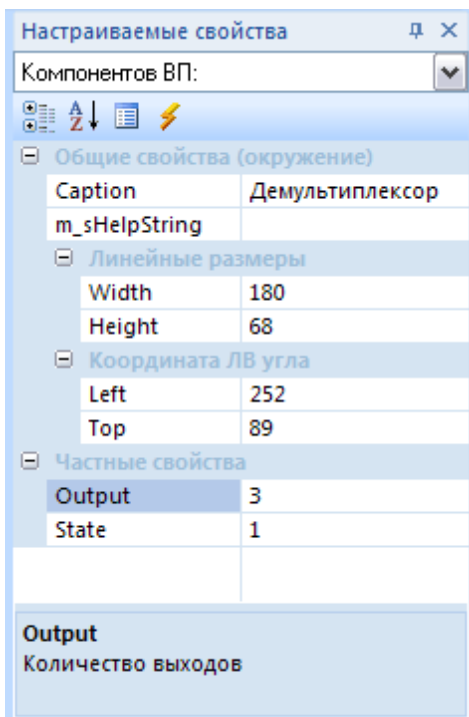


Рисунок 2.2.8. Настраиваемые свойства демultipлексора.

Выставим свойство "Output" равное 3 (см.рис.2.2.8.). Подсказка внизу окна настраиваемых свойств сообщает нам о том, что свойство "Output" отвечает за количество выходов демultipлексора. После изменения свойств у компонента добавится два выхода (см.рис.2.2.9.).

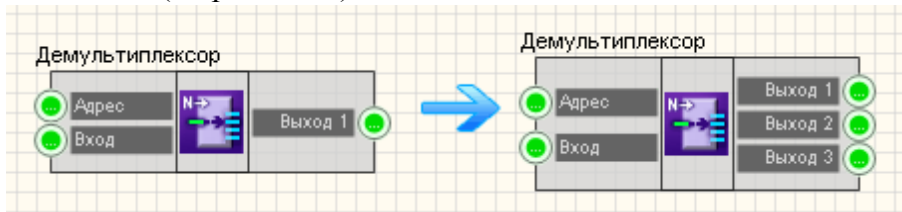


Рисунок 2.2.9. Изменения демultipлексора.

Проведем связи как показано на рисунке 2.2.10.

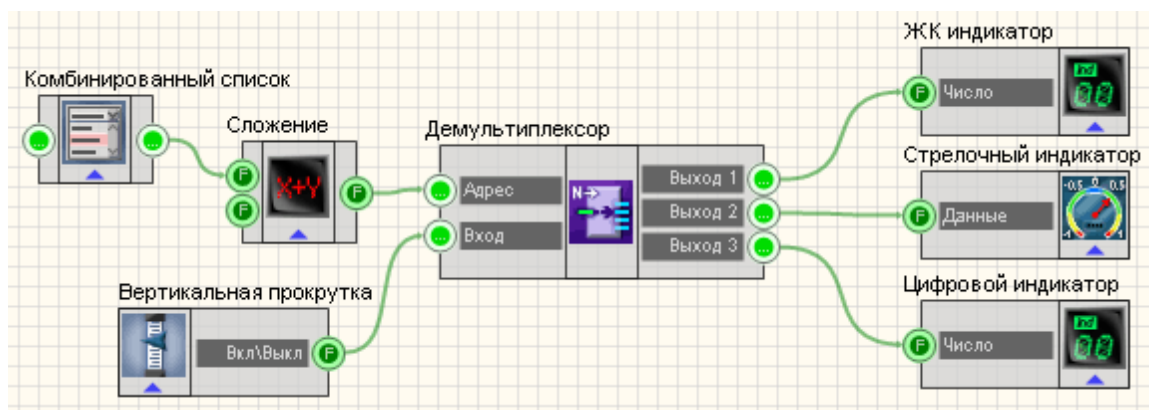
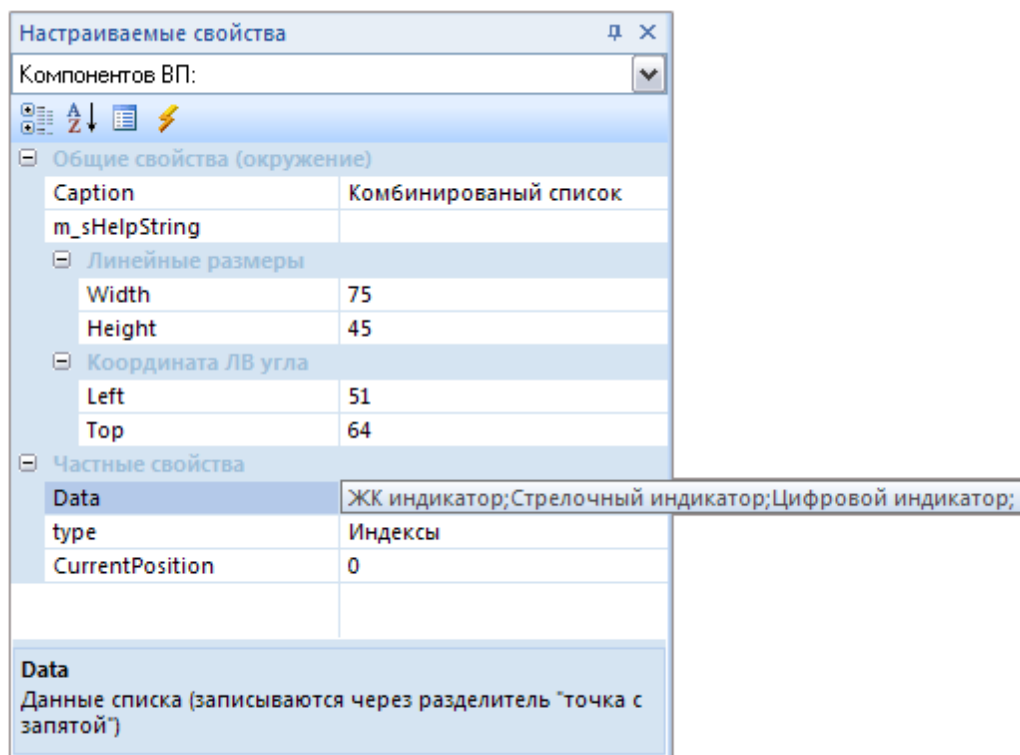


Рисунок 2.2.10. Создание связей.

Настроим оставшиеся компоненты (см.рис.2.2.11)



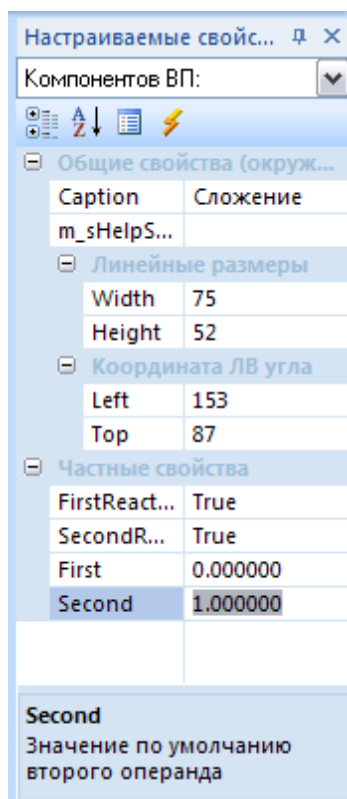


Рисунок 2.2.11. Свойства компонентов.

В свойствах комбинированного списка прописываем пункты списка через точку с запятой, и свойство `type` выставляем значение "Индексы". Таким образом при выборе одного из трех пунктов списка компонент будет передавать на выход индекс выбранного пункта. Пункты списка нумеруются по порядку, начиная с 0. Именно поэтому в проекте используется компонент "Сложение". Демультимплексор получает на вход номер выходного контакта, на который следует передавать данные, и нумеруются они начиная с 1. Поэтому чтобы значение переданное списком соответствовало выходу на демультимплексоре используется компонент "Сложение" который прибавляет единицу к полученному индексу списка. Настроим компонент сложение, а именно свойство "Second" - значение на втором входе - фактически второе слагаемое, первое слагаемое - индекс элемента списка.

Перейдем в интерфейс оператора и настроим внешний вид проекта. Для этого убедимся, что на ленте включена вкладка "Главная панель" и нажмем на этой панели кнопку "Интерфейс оператора".

После перехода можно обнаружить все компоненты "в куче". Необходимо "расташить" их, как удобно для восприятия пользователя или как показано на рисунке 2.2.6. В "Интерфейс оператора" можно изменить линейные размеры, например у компонента ЖК индикатор и размер шрифта (свойство `TextSize`), чтобы шрифт умещался в уменьшенный ЖК индикатор. Подробно о настройке внешнего вида [см.гл.11](#)<sup>[60]</sup>.

Теперь сохраним проект с помощью главного меню и запустим нажатием кнопки

"Запустить проект" на главной панели или нажатием кнопки F5 на клавиатуре.

Теперь, выбрав из списка один из индикаторов, можно перемещать указатель прокрутки и наблюдать результат. При перемене индикатора на предыдущем индикаторе сохраняется последнее значение, это обусловлено особенностью работы SCADA ZETView, так как предполагается, что система работает с сигналом в реальном времени от  $-\infty$  до  $+\infty$ . И ЖК, и цифровой индикаторы поменяют значение при перемене значения на входе. Если работа шла с реальным или сгенерированным сигналом, при его отсутствии индикатор показывал бы 0.

Данный пример непосредственно в ZETView

## Глава 3. Расширяем возможности настройки

В предыдущей главе рассматривался простейший пример, где активно использовалась настройка свойств компонентов. Обилие настроек весьма удобно, но как быть если свойства одного компонента напрямую зависят от состояния другого. В проектах чаще всего используется достаточно сложный компонент сценарий, который будет рассмотрен в отдельной главе. Также для этих целей служит универсальный компонент «Менеджер свойств» позволяющий добавить компоненту недостающие контакты, но обладающий меньшими возможностями.

Многие компоненты имеют специальный вход для подключения менеджера свойств, который располагается внизу компонента в виде треугольника.

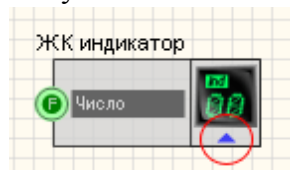


Рисунок 2.3.1. Вход менеджера свойств

Если ко входу подключить менеджер свойств, то можно дать возможность оператору настраивать свойства компонента напрямую.

Обратим внимание, что при подключении к разным компонентам, менеджер свойств принимает разный вид:

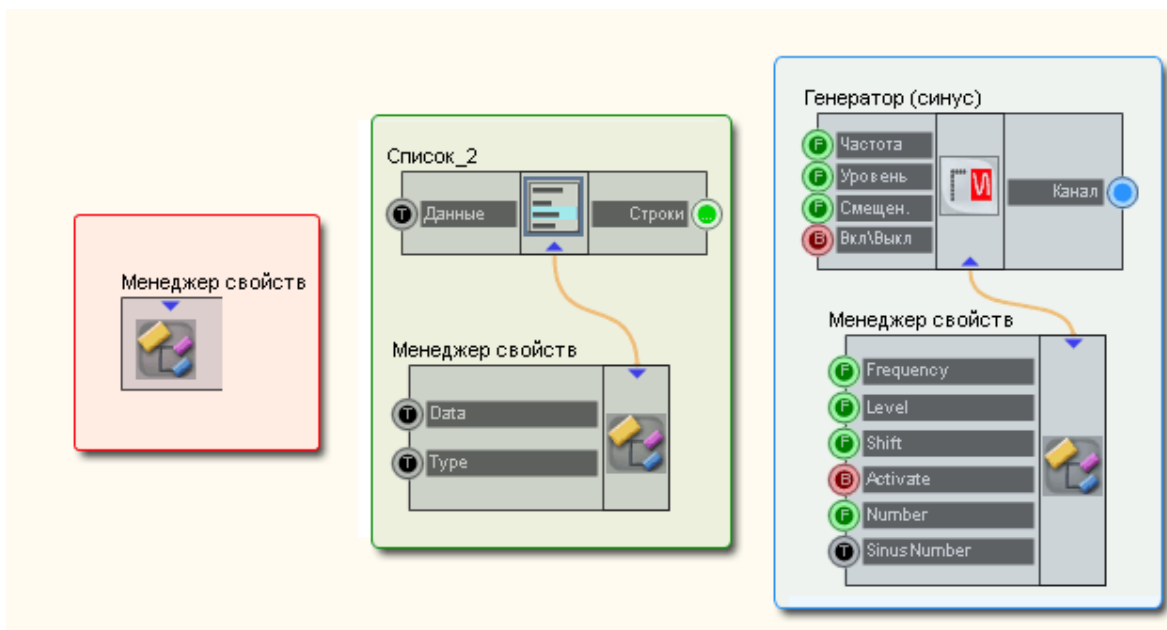


Рисунок 2.3.2. Различные формы менеджера свойств

Менеджер свойств дает доступ оператору к свойствам компонента, которые не были доступны через входы и их нужно было прописывать вручную.

Например на рисунке 2.3.3 показан обычный ЖК -индикатор, слева с помощью кнопок мы можем настроить его внешний вид.



Рисунок 2.3.3. Использование менеджера свойств для настройки индикатора

Соберем проект в SCADA ZETView.

Перенесем в рабочую область следующие компоненты:

Группа "Выбор" -> Горизонтальная прокрутка

Группа "Выбор" ->Кнопка без фиксации - 2шт

Группа "Выбор" ->Кнопка с фиксацией

Группа "Выбор" ->Селектор №1

Группа "Разное" ->Выбор цвета - 2шт



Группа "Выбор" -> Менеджер свойств  
 Группа "Индикация" -> ЖК индикатор

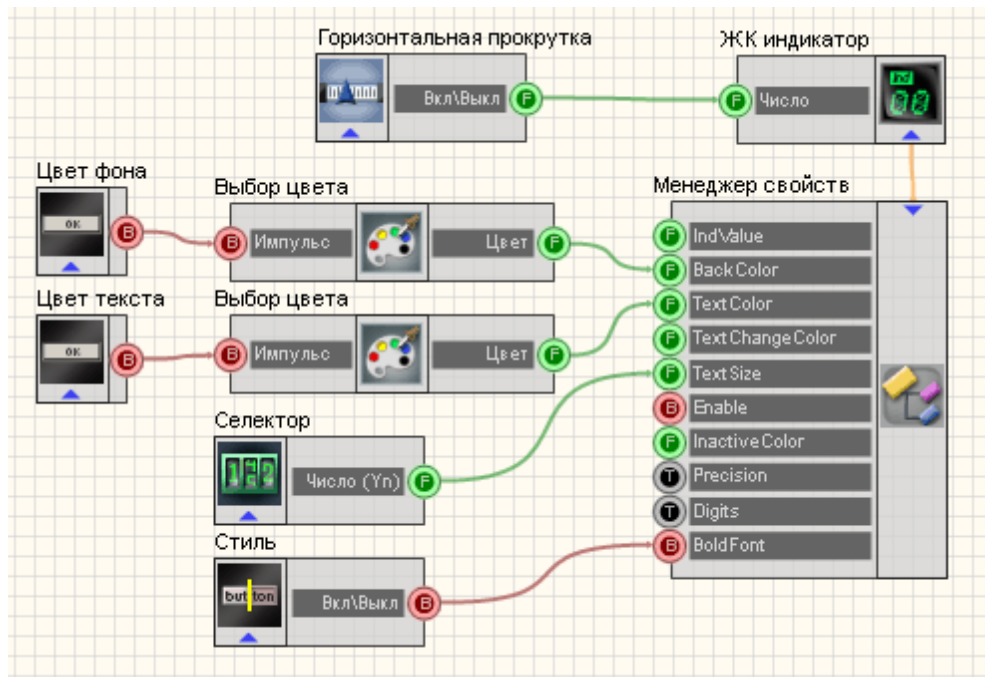


Рисунок 2.3.4. Проект в SCADA ZETView

На вход ЖК индикатора подключим горизонтальную прокрутку и менеджер свойств к специальному входу. Как только менеджер свойств будет подключен он изменит свой внешний вид и приобретет несколько входов.

К входам «BackColor» и «TextColor» нужно подключить по компоненту «Выбор цвета». К каждому компоненту «Выбор цвета» подключается «Кнопка без фиксации». Селектор №1 подключается к входу «TextSize», а кнопка с фиксацией на вход «BoldFont».

Далее настроим компоненты.

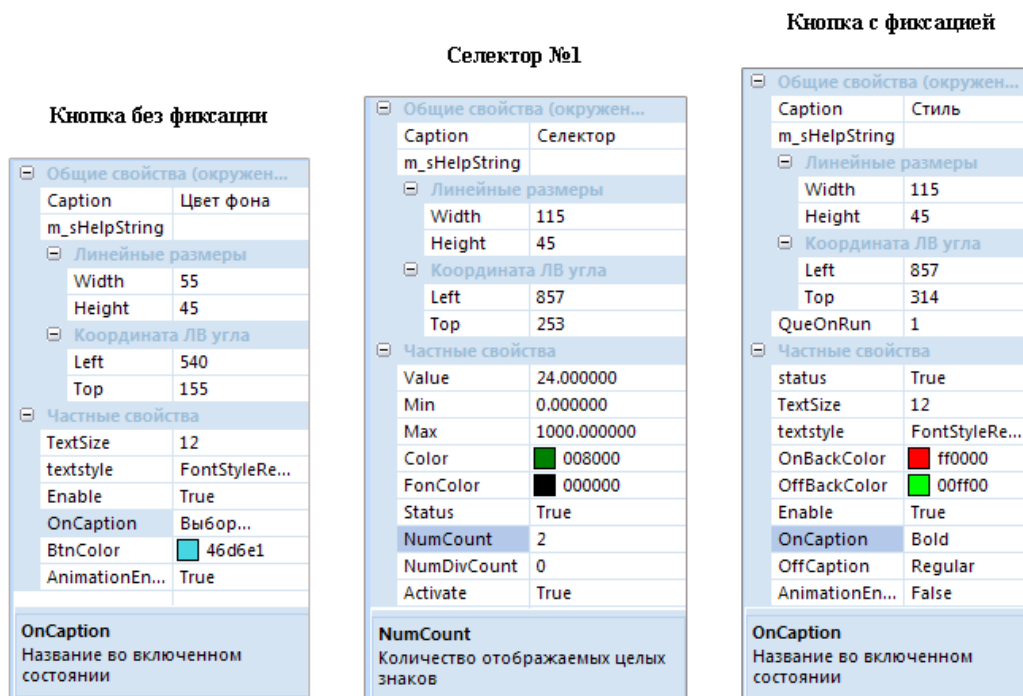


Рисунок 2.3.5 Свойства компонентов

После того, как соединения проведены и компоненты настроены можно приступить к настройке проекта в режиме оператора.

Как можно заметить на рисунке 2.3.2 напротив кнопок и индикаторов есть комментарии. Для того, чтобы поместить комментарий используется компонент «Статический текст Windows стиль» из группы текст. Данный компонент нужно использовать только в режиме оператора, так как он не имеет внешнего вида в режиме проектирования.

Данный пример непосредственно в ZETView

## Глава 4. Работа с текстом

Вывод и обработка текстовой информации - одна из важных функций взаимодействия с пользователем. В SCADA ZETView с помощью компонентов для работы с текстом можно информировать пользователя о процессе выполнения проекта, выдавать предупреждения и сообщения об ошибках, вводить параметры и значения, записывать информацию в таблицы.

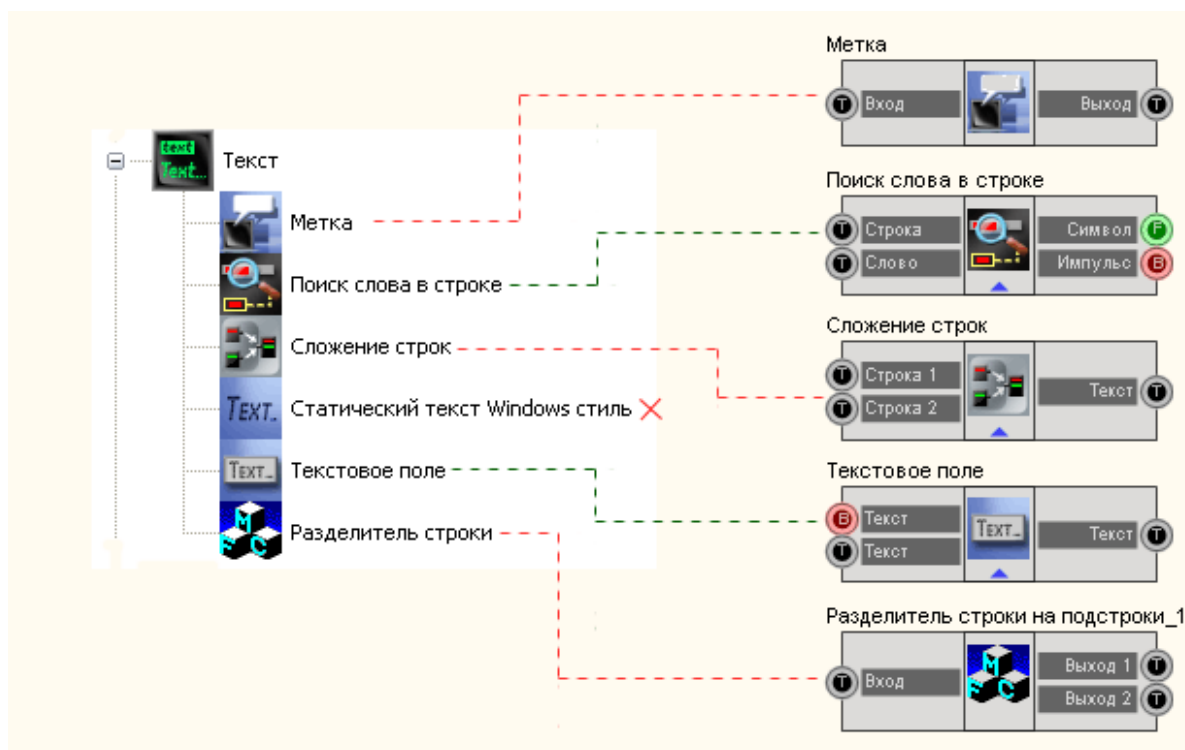


Рисунок 2.3.1. Группа "Текст".

Группа "Текст" содержит основные компоненты для работы с текстовой информацией. Эти компоненты часто используются в проектах для вывода уведомлений, получения и обработки текстовых данных.

Для ввода текстовой информации предусмотрен компонент «Текстовое поле» где пользователь может ввести текстовые данные и затем передать их последующему компоненту, при каждом их изменении либо при получении управляющего импульса на специальный вход (например, при подключении кнопки к этому входу, компонент передаст введенную строку при ее нажатии).

Также в группе «Текст» имеются инструменты обработки текстовой информации. Разделитель строки делит входную строку на две и более, по указанному символу-разделителю. Количество выходных строк также указывается в настройках

Компонент «Метка» используется для вывода текстовой строки. Если необходимо отобразить массив данных, для этого используется компонент «Таблица данных»

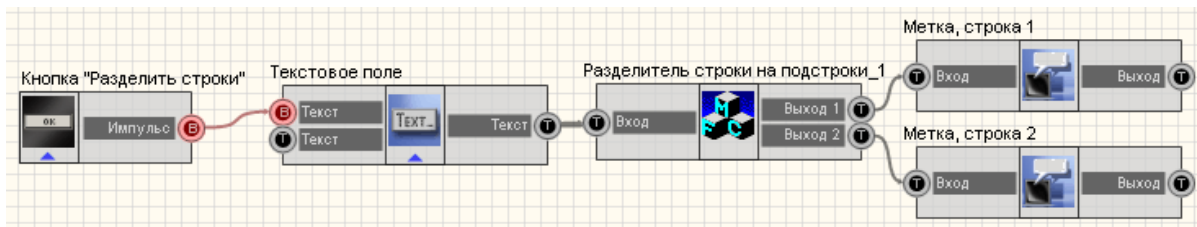
Компонент «Статический текст Windows стиль», который не доступен в режиме проектировщика, виден только в режиме оператора. Данный компонент играет роль «комментария». С помощью него можно сделать некоторые текстовые пометки для оператора. Этот компонент уже упоминался в предыдущей главе.

Соберем проект в SCADA ZETView.

Перенесем в рабочую область следующие компоненты:

Группа "Кнопки" -> Кнопка без фиксации

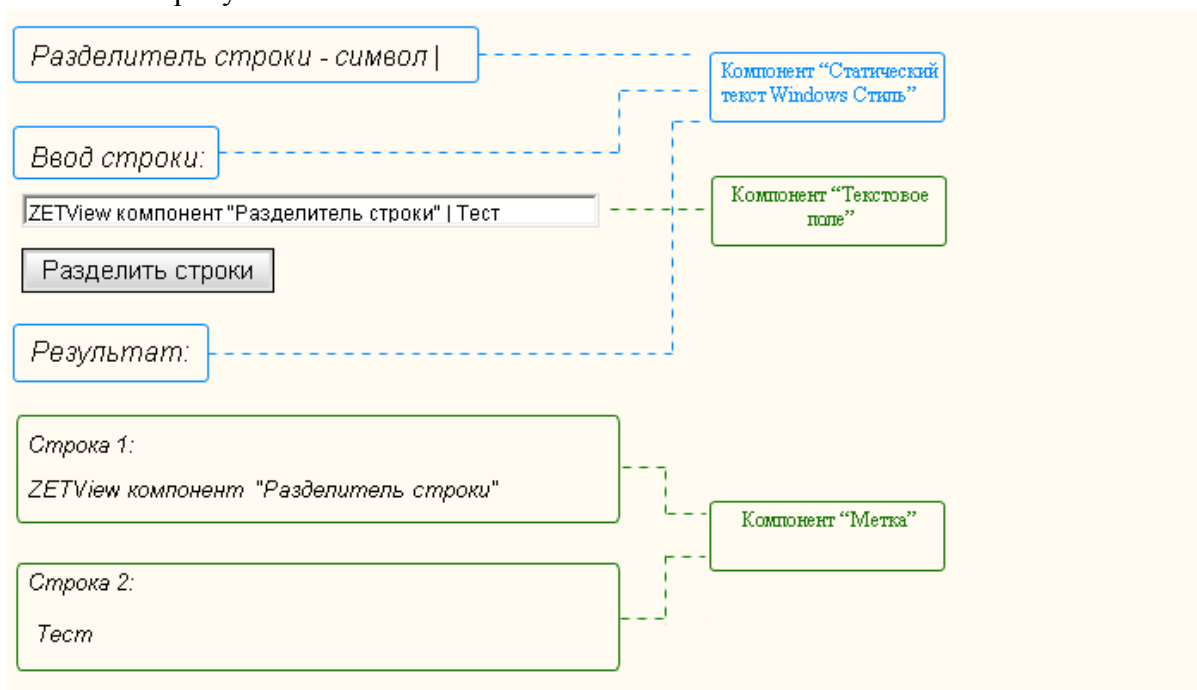
- Группа "Текст" -> Текстовое поле
- Группа "Текст" -> Разделитель строки
- Группа "Текст" -> Метка 2шп



Выставим свойства компонентов:

- Кнопка без фиксации - Свойство OnCaption: Разделите строки;
- Разделитель строки - Свойство Delimiter: поставить значок "|" (без кавычек)

Далее переключимся в режим оператора и расположим компоненты как показано на рисунке.



Данный пример непосредственно в ZETView

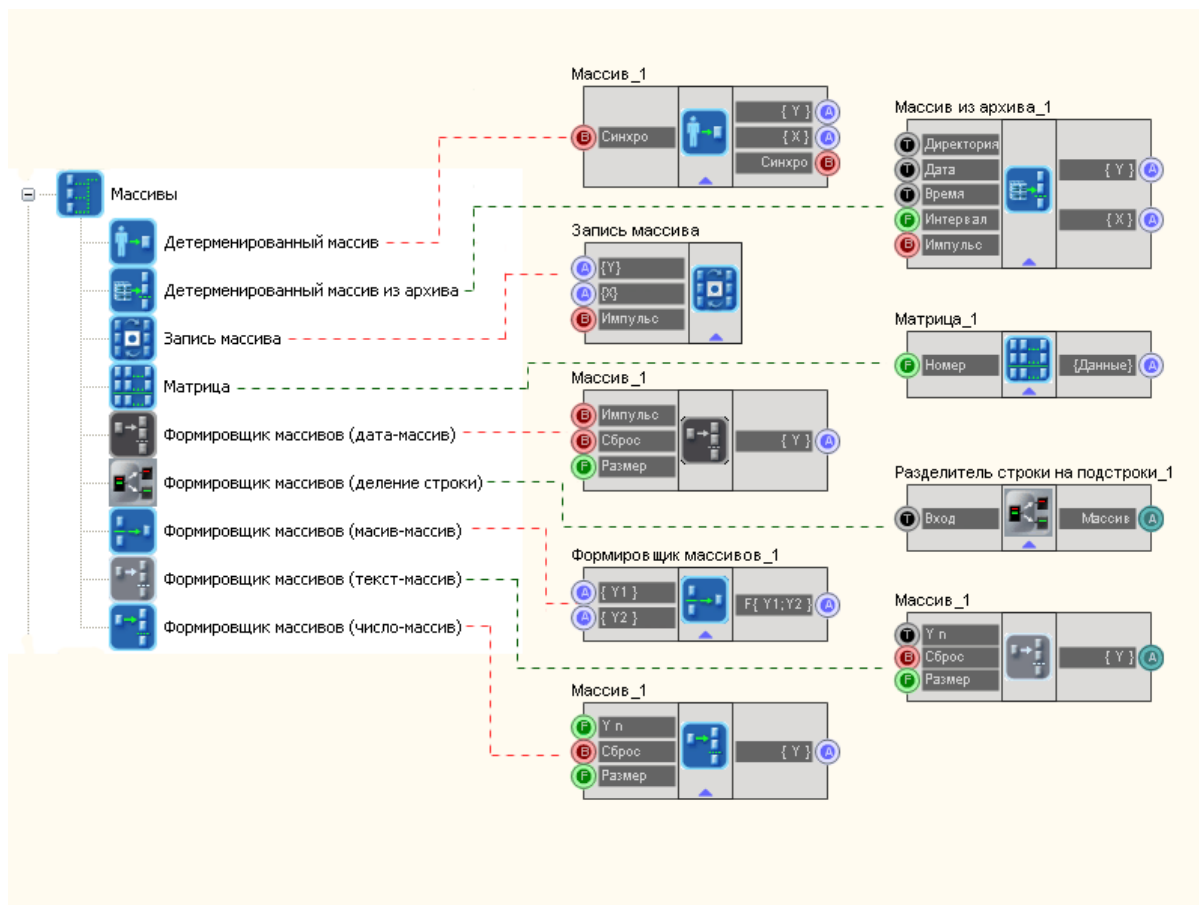
## Глава 5. Основы использования массивов

Многие действия в SCADA сопряжены с использованием массивов. Массив является наиболее удобным типом данных для хранения информации о измеренных значениях сигнала в режиме реального времени, любого числового или символьного ряда. Многие компоненты в SCADA ZETView используют массивы как входной параметр. На данном этапе будем рассматривать массивы абстрактно, используя простые числа, после изучения работы с виртуальными приборами, массивы будут рассмотрены снова, применительно к обработке сигналов в реальном времени.

Для работы с массивами предусмотрено три группы компонентов:

- 1 – Массивы
- 2 – Массив-Значение
- 3 – Массив-Массив

Название группы компонентов определяет входные и выходные параметры компонентов в определенной группе. В группе «Массивы» содержатся компоненты, отвечающие за формирование и запись массивов. В группе «Массив-Значение» содержатся компоненты, производящие определенные действия с входным массивом и выдающие число, такие как, например, сумма элементов массива. В группе «Массив-Массив» содержатся компоненты, получающие на вход массив, и соответственно, передающие массив с выходного контакта.



Рассмотрим группу «Массивы» более подробно. В данной группе расположены инструменты для формирования массивов различного типа, представленные компонентами типа «Формировщик массивов». Повсеместно используется «Формировщик массивов (число-массив)». Для многих конструкций этот компонент является обязательным, так как, при измерении характеристик сигнала, виртуальные приборы передают число, а компоненты, обрабатывающие полученные данные, зачастую требуют получать их, как массив.

Как видно из рисунка, в SCADA ZETView доступно несколько разных видов формировщика массивов. Все они помещают полученные данные в массив определенного размера или динамический массив, размер которого считается исходя из количества элементов, полученных на данном момент, формировщики массивов с подобной конфигурацией широко используются в проектах при проведении измерений. Пользователю доступны:

Формировщик массивов (число-массив) - этот компонент получает на вход числовые значения а с выхода передает числовой массив.

Формировщик массивов (текст-массив) - компонент получает на вход строки и формирует из них строковый массив, часто используется для вывода результатов в таблицу. Так как для вывода результата в таблицу, на вход подается строковый массив.

Формировщик массивов (массив-массив) - компонент формирует один массив из двух входящих, используя различные способы: поэлементное сложение, поэлементное среднее, выбор минимального или максимального значения и другие.

Формировщик массивов (деление строки) - компонент работает аналогично уже известному нам компоненту "Разделитель строки", с тем отличием, что пользователю не нужно указывать точное количество выходных контактов, на которые будут подаваться полученные строки, здесь все полученные строки будут записываться в текстовый массив.

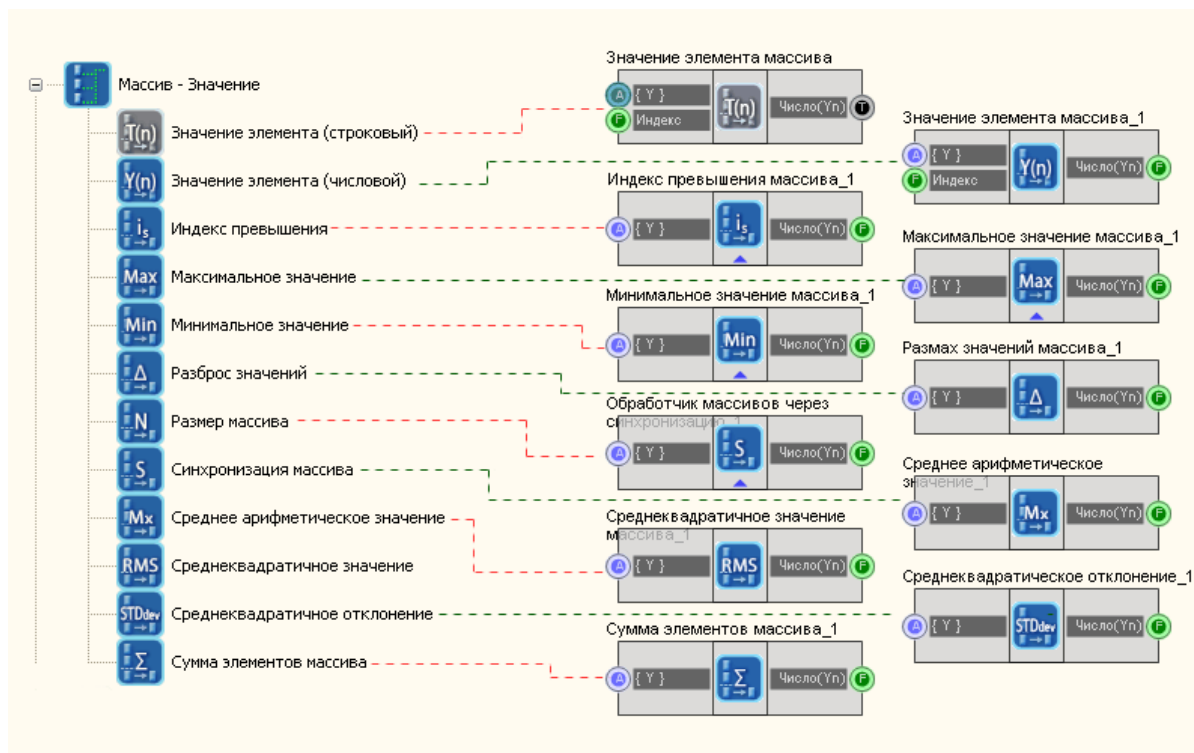
Формировщик массивов (дата-массив) - записывает в массив текущую дату и время при получении импульса на вход. Компонент используется для индикации измерений и формирования отчетов, содержащих не только сами значения но и время, когда измерения были сняты.

Матрица - компонент, выводящий указанную строку или столбец из заранее записанной в него матрицы, используется как многомерный массив.

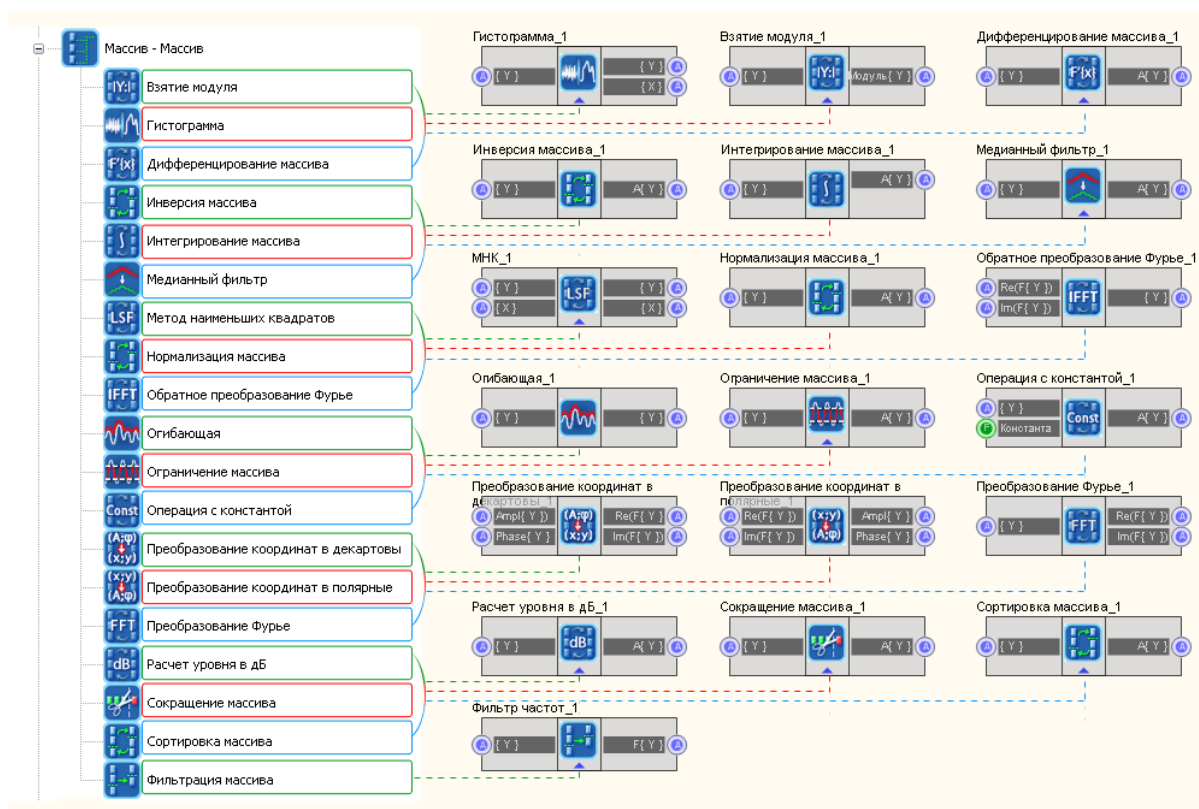
Детерминированный массив - автоматический формировщик массивов, исходя из типа формулы (константа, линейная и квадратичная) и заданных параметров.

Детерминированный массив из архива - воспроизводит массив, с ранее записанного сигнала, с помощью компонента "Запись каналов".

Запись массива - компонент, записывающий полученный массив в файл, при получении импульса на вход.



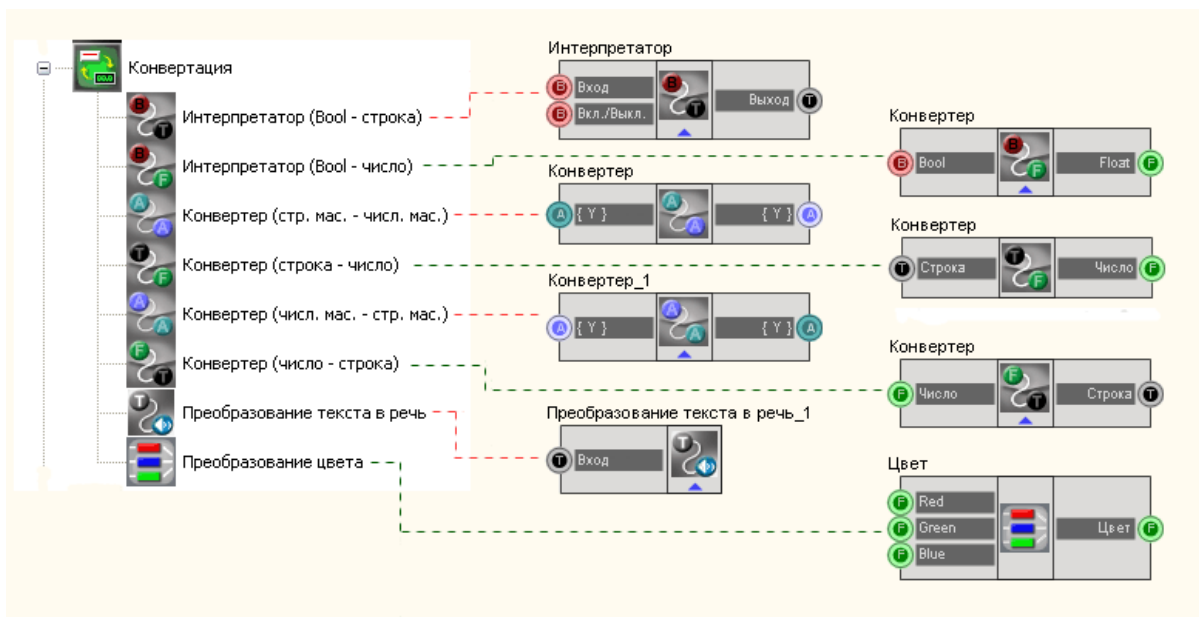
В группе массив-значение содержатся элементы обрабатывающие полученный массив и выдающие необходимое значение. Посредством компонентов из данной группы можно ограничить массив сверху или снизу, вычислить значение элемента в массиве или сумму элементов.



## Глава 6. Конвертация

Как известно из предыдущей главы, в SCADA ZETView существуют различные типы данных. В большинстве, каждый компонент принимает и передает определенный тип данных. Во многих случаях необходимо преобразование типов данных. Типичный пример для этого вывод числовой информации в таблицу. Данный компонент принимает текстовую информация на входной контакт, поэтому, чтобы вывести в таблицу число, необходимо сначала представить его в виде текста. Для подобных операций используются компоненты из группы «Конвертация». В этой группе находятся все необходимые компоненты для конвертирования всех имеющихся типов данных в SCADA ZETView.

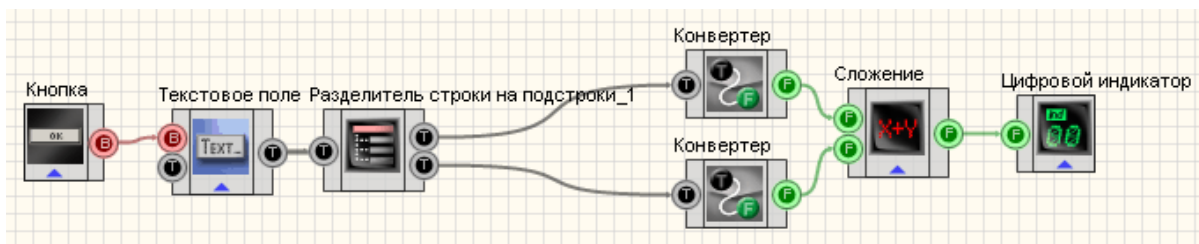




Соберем проект в SCADA ZETView. В этом проекте пользователю предложено ввести текстовую строку вида 1+2. Для упрощения проекта будем использовать только знак «+». Далее из строки выделяются цифры и преобразуются из строкового типа данных в числовой, после чего вычисляется сумма введенных чисел.

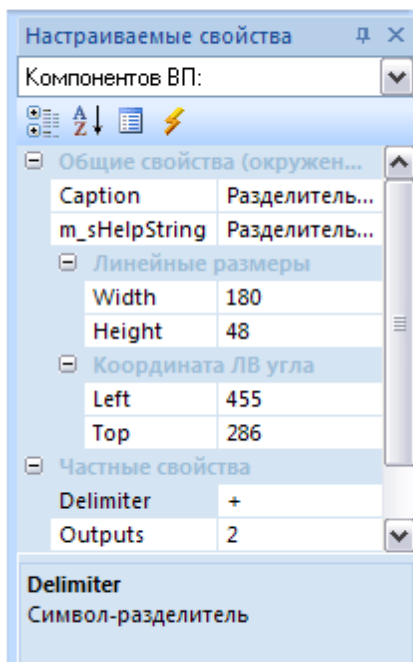
Поместим в рабочей области компоненты как показано на рисунке 2.3.2.

- Группа "Текст" => компонент "Текстовое поле"
- Группа "Текст" => компонент "Разделитель строки"
- Группа "Кнопки" => компонент "Кнопка без фиксации"
- Группа "Конвертация" => компонент "Конвертер(строка-число)" - 2шт.
- Группа "Арифметика" => компонент "Сложение"
- Группа "Индикация" => компонент "Цифровой индикатор"

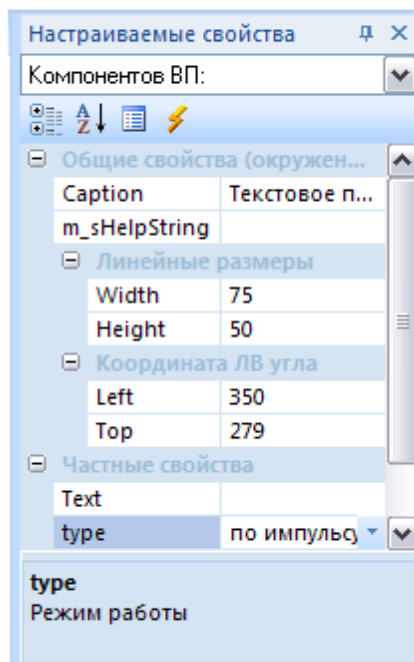


Как можно заметить, на рисунке компоненты изображены в "свернутом" виде (чтобы свернуть или развернуть компонент достаточно сделать двойной щелчок левой кнопкой мыши по компоненту). Необходимо настроить свойства компонента "Разделитель строки", где выставляется символ-разделитель "+", чтобы компонент распознал записанную пользователем строку. Также настраивается компонент "Текстовое поле" таким образом, чтобы он передавал значение при получении управляющего импульса на вход.

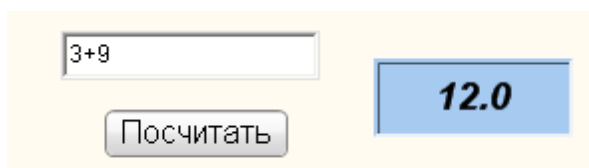
Разделитель строки



Текстовое поле



При получении управляющего импульса на вход, компонент «Текстовое поле» передает введенную пользователем строку следующему компоненту «Разделитель строки». В свойствах компонента «Разделитель строки», символом, указывающим на разделение строк, установлен знак «+». Поэтому если ввести строку вида 1+2 мы получим две текстовых строки: «1» и «2». Нельзя сложить текстовые переменные «1» и «2» и получить результат, для компьютера это выглядит как попытка сложить букву "А" и "Б", так как эти цифры воспринимаются машиной как строка и не отличаются по свойствам от букв. Поэтому символы нужно перевести в числовой формат. Далее используется компонент «Конвертер (строка-число)». С выходов конвертеров уже поступают два числа, которые передаются далее на компонент «Сложение», после чего сумма выводится на цифровой индикатор.



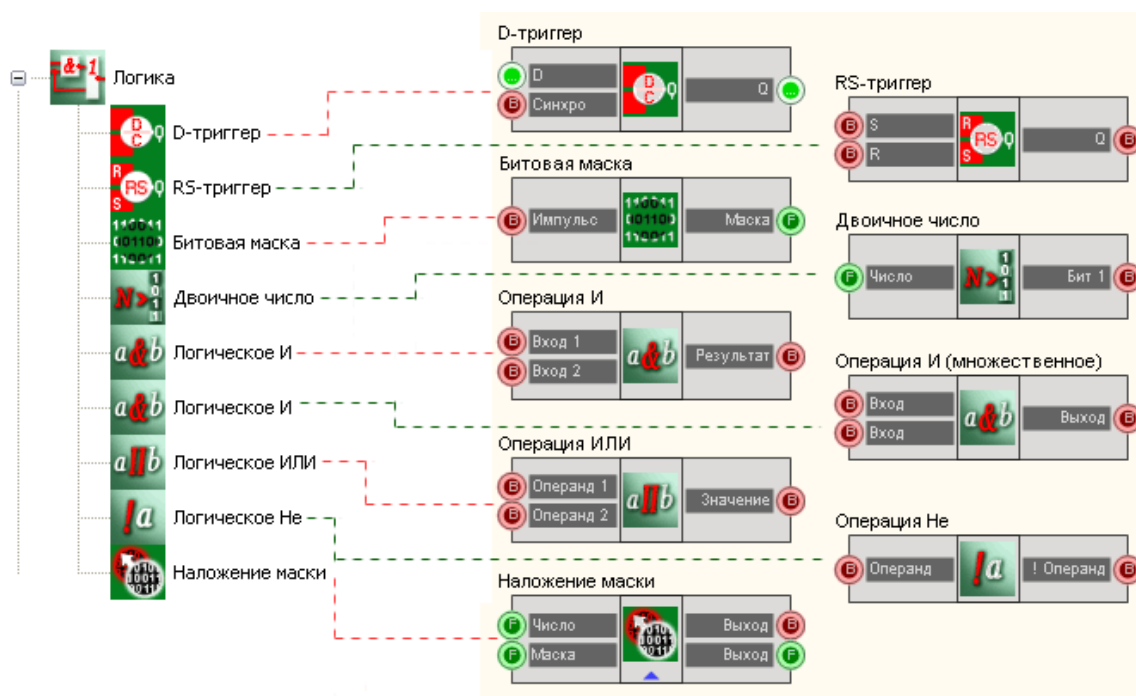
Данный пример непосредственно в ZETView

В представленном примере показана работа конвертера строка-число. Также в

группе конвертация имеются другие конвертеры, позволяющие конвертировать строку в число, строковые и числовые массивы. В случае когда, например, конвертируется строковый массив в числовой или строка в число и конвертация не возможна, конвертер просто проигнорирует неконвертируемое значение. Также можно найти в группе интерпретатор Bool-строка и Bool-число. Интерпретатор Bool-строка выводит 0 или 1 - в зависимости от логического уровня на входе. Интерпретатор Bool-строка выводит одну из двух строк, прописанных пользователем в настройках компонента, в зависимости от логического уровня на входе.

## Глава 7. Арифметика и логика

В проектах SCADA ZETView управление осуществляется с помощью компонентов группы логика и программируемых компонентов. Программируемый компонент будет рассмотрен в следующих главах. В группе «Логика» содержатся компоненты, реализующие такие логические операции, как «Операция И», «Операция ИЛИ», «Операция НЕ», «Операция И (множественное)» (отличается лишь тем, что может принимать от двух и более значений на вход). Также имеются D- и RS-триггеры, который передает полученную информацию при получении импульса на вход «Синхро». RS – триггер, передающий логическую переменную, зависящую от комбинации на входе.



## Глава 8.Отладка

Отладка – трудоемкий процесс, который неизбежен при создании проектов.

Просмотр запущенных программ возможен через unit в проекте для отладки в ZETView.exe. Юниты становятся видимыми при вводе команды shu1 и скрываются shu0.

Команда вводится с клавиатуры после щелчка мыши по любой из страниц ZETView. Для того чтоб Юниты появились, они должны быть запущены, часто запуск юнитов происходит после старта проекта.

Так же добавлено в SCADA-компоненты;

- Вольтметр постоянного тока;
- Вольтметр переменного тока;
- Генератор сигналов;
- Синхронный генератор;
- Виброметр;
- Долектавный спектр.

# Часть 3. Основы создания проектов в SCADA ZetView

Enter topic text here.

## Глава 1. Часть 2. Основы создания проектов в SCADA ZetView

# Часть 4. Компоненты ZETView

## Глава 1. Общие свойства (окружение)

**Общие свойства (окружение)** компонентов - это набор графических и идентификационных свойств, которыми обладают все компоненты SCADA ZETView.

➤ Общие свойства (окружение):

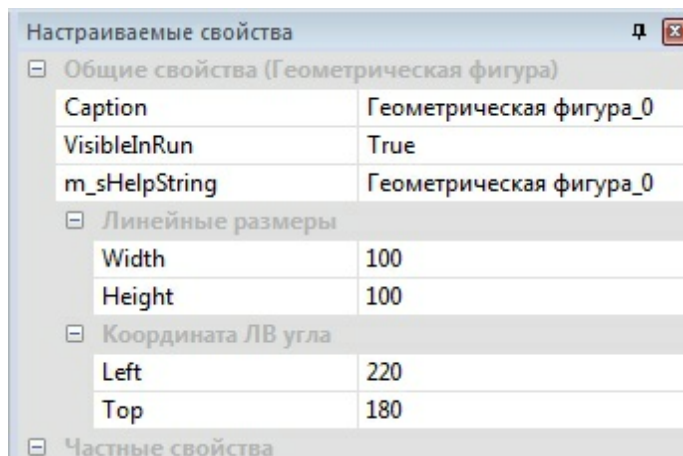
- **Caption** - название компонента (это имя используется при написании скриптов, также при межстраничной связи).
- **VisibleInRun** - видимость компонента при запущенном проекте.
- **m\_sHelpString** - строка идентификатора (используется в адресации канала с компонентом).

➤ Линейные размеры:

- **Width** - ширина компонента.
- **Height** - высота компонента.

➤ Координаты левого угла:

- **Left** - координаты X левого верхнего угла.
- **Top** - координаты Y левого верхнего угла.

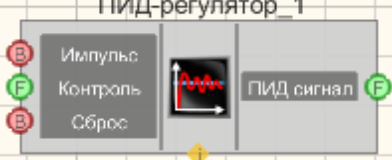


## Глава 2. Автоматизация

### 2.1. ПИД-регулятор

**ПИД-регулятор** - компонент. Предназначен для создания автоматизированных систем управления на базе модулей АЦП-ЦАП, например модуль «SigmaUSB», для поддержания контролируемой величины, равной заданному значению.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Импульс - контакт, при подаче на который импульса происходит запуск/остановка выполнения регулирования.
- Контроль - контакт обратной связи регулируемого компонента.
- Сброс - контакт, при подаче на который импульса, происходит сброс интегральной составляющей (сумма накопленных ошибок становится равна 0).

##### ➤ Выходные

- ПИД-сигнал – контакт с которого отправляется на регулируемый компонент управляющее воздействие.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (0) - текущее значение.
- Enable (true) - включение/отключение компонента
- PropKcoef (1) - коэффициент при пропорциональной составляющей (противодействует отклонению регулируемой величины от заданного значения)
- IntegrKcoef (0) - коэффициент при интегральной составляющей (учет предыдущих значений, способствует устранению статической ошибки)
- DiffKcoef (0) - коэффициент при дифференциальной составляющей (обеспечивает быструю реакцию на отклонение от заданной величины)
- MaxCurrentValue (0) - максимальное значение управляющего сигнала. Если значение 0, то не используется.

- MinCurrentValue (0) - минимальное значение управляющего сигнала. Если значение 0, то не используется.



## Пример

### Проект в SCADA ZETView



### Результат работы проекта

Вкл

Sig\_1\_1  
на вольтметре

0.1

на выходе регулятора

989.8

[Посмотреть пример непосредственно в ZETView](#)

## 2.2. ПИД-регулятор из таблицы

**ПИД-регулятор из таблицы** - компонент. Предназначен для регулирования уровня сигнала в соответствии с заданным профилем в таблице.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------





### Параметры:

#### ➤ Входные:

- Импульс - контакт, при подаче на который импульса происходит запуск/остановка выполнения регулирования.
- Значения - контакт, который подается массив значений.
- Время - контакт, который подается массив времени, с.
- Контроль - контакт обратной связи регулируемого компонента.
- Сброс - контакт, при подаче на который импульса, происходит сброс интегральной составляющей (сумма накопленных ошибок становится равна 0).

#### ➤ Выходные

- ПИД-сигнал - контакт с которого отправляется на регулируемый компонент управляющее воздействие.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

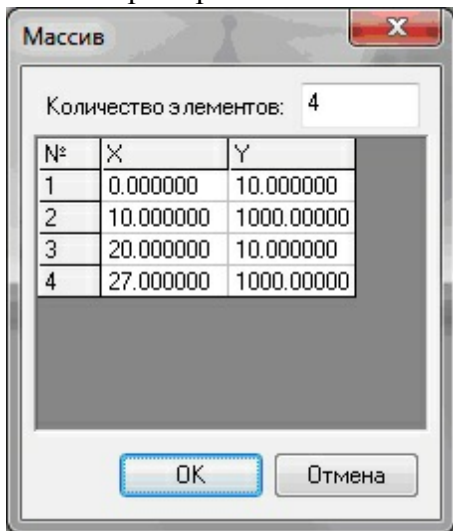
- Enable (true) - включение/выключение ПИД-регулятора.
- PropKoeff (1) - коэффициент при пропорциональной составляющей (противодействует отклонению регулируемой величины от заданного значения)
- IntegrKoeff (0) - коэффициент при интегральной составляющей (учет предыдущих значений, способствует устранению статической ошибки)
- DiffKoeff (0) - коэффициент при дифференциальной составляющей (обеспечивает быструю реакцию на отклонение от заданной величины)
- MaxCurrentValue (0) - максимальное значение управляющего сигнала. Если значение 0, то не используется.
- MinCurrentValue (0) - минимальное значение управляющего сигнала. Если значение 0, то не используется.
- ProcessVisible (false) - включение/отключение отображения процесса регулирования.

### Дополнительные сведения:

Данный компонент удобно использовать совместно с компонентом

«Детерминированный массив» при задании в нем источника данных «из таблицы».

Например:

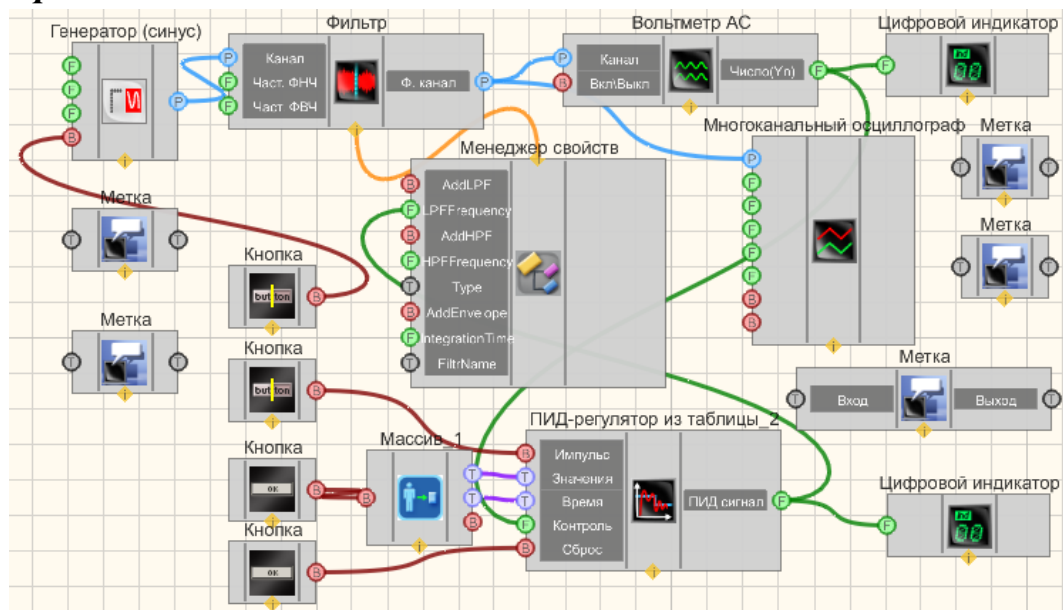


В данной таблице столбец X используется как массив времени (время указывается в секундах), столбец Y – массив значений. Первое значение массива времени всегда должно быть равным 0, а первое значение массива данных – первое управляющее воздействие, отправляемое управляемому объекту.



## Пример

### Проект в SCADA ZETView



### Результат работы проекта

1. Включить генератор; 2. Инициализировать регулятор; 3. Включить регулятор.

Генератор

Регулятор

инициализировать регулятор



на вольтметре

на выходе регулятора

Время 0.172620 с    filtr2 = 2317.13 мВ



## 2.3. Синхронизация по GPS

<Раздел находится в разработке>

**Синхронизация по GPS** - компонент. Предназначен для синхронизации удаленных объектов между собой, а также для привязки синхронизации к шкале единого времени.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
<p>Синхронизация по GPS</p> 	<p>Не имеет</p>

**Параметры:**

➤ Входные:

Не имеет.

➤ Выходные

- Год - выходное значение данных год.
- Месяц - выходное значение данных месяц.
- Число - выходное значение данных число.
- Час - выходное значение данных час.
- Минута - выходное значение данных минута.
- Секунда - выходное значение данных секунда.
- Широта - выходное значение данных широта.
- Долгота - выходное значение данных долгота.
- Скорость - выходное значение данных скорость.
- Спутники - количество спутников.
- Сдвиг - сдвиг по времени

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

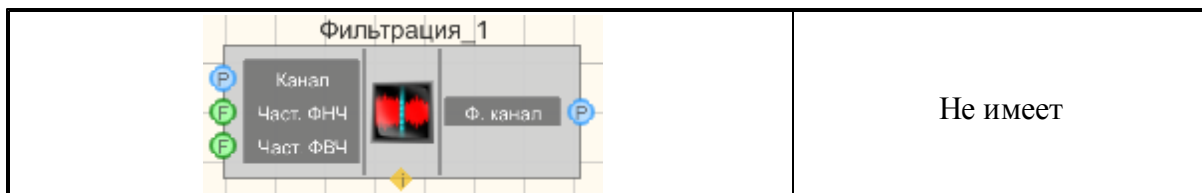
- speed (57600) - скорость приема данных NMEA потока, бит/с:
  1. 57600 бит/с
  2. 38400 бит/с
  3. 19200 бит/с
  4. 9600 бит/с
  5. 4800 бит/с
- Synchro (false)- разрешение синхронизации.
- Lead (1) - опережение запуска, с
- inputport (COM2) - последовательный порт для обмена с GPS-приемником
- outputport (отсутствует)- последовательный порт для выхода данных GPS

## 2.4.Фильтрация

**Фильтрация** - компонент. Создает дополнительные виртуальные каналы для обработки сигналов Все сигналы - реальные и виртуальные имеют внутреннюю синхронизацию, что позволяет проводить их совместную обработку. Предназначен для достоверного оценивания какого-либо параметра сигнала, например, уровня, частоты, коэффициента корреляции с другим сигналом.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входящие:

- Канал - фильтруемый канал.
- Част.ФНЧ - частота среза ФНЧ, Гц
- Част.ФВЧ - частота среза ФВЧ, Гц

#### ➤ Выходные

- Ф.канал - выходной канал фильтра

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- AddLPF (false) - разрешения работы фильтра нижних частот
- LPFFrequency (1000) - частота среза фильтра нижних частот, Гц
- AddHPF (false) - разрешения работы фильтра верхних частот
- HPFFrequency (100) - частота среза фильтра нижних частот, Гц
- type (линейный) - тип фильтра:
  1. линейный
  2. дифференцирующий 1-го порядка
  3. дифференцирующий 2-го порядка
  4. интегрирующий 1-го порядка
  5. интегрирующий 2-го порядка
- AddEnvelope (false) - разрешения работы детектора огибающей
- IntegrationTime (0) - время интегрирования, мс
- FiltrName - имя канала фильтра.



### Пример

**Пример** фильтрация: (находится в Примере компонента [ПИД-регулятор\\_из\\_таблицы](#)<sup>[94]</sup>)

## Глава 3. Анализ сигналов

### 3.1. Анализ нелинейных искажений

**Анализ нелинейных искажений** - компонент. Предназначен для автоматического измерения коэффициента нелинейных искажений и среднеквадратических значений напряжения исследуемых сигналов, поступающих на входные каналы анализаторов спектра.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Канал - входной измерительный канал;
- Вкл \ Выкл - включение\выключение канала анализа нелинейных искажений;
- Запись - запись в файл.

##### ➤ Выходные

- Значение (Hn) - выходные значения гармоники;
- Значение (Cn) - выходные значения коэффициента гармоники;
- Канал - канал нелинейных искажений.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

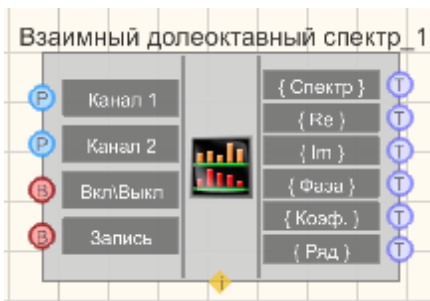
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- AverageTime (1) - время усреднения, с.
- freqrange (50000) - частотный диапазон, Гц.
- calcType (линейное) - типа представления расчета коэффициента нелинейных искажений:
  1. линейный масштаб (в процентах)
  2. логарифмический масштаб (в децибелах)
- FileName - имя файлов для записи результатов

## 3.2. Взаимный долеоктавный спектр

**Взаимный долеоктавный спектр** - компонент. Предназначен для частотного анализа сигналов. Возможность получения максимальных, усредненных, минимальных и мгновенных спектров и сравнение их с заданным спектром (нормой) позволяет легко определить различие между заданным и реальным уровнем спектров.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал 1 - первый измерительный канал.
- Канал 2 - второй измерительный канал
- Вкл/Выкл - включение/выключение спектра.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

#### ➤ Выходные

- {Спектр} - текущий мгновенный взаимный долеоктавный спектр.
- {Re} - действительная часть взаимного долеоктавного спектра.
- {Im} - мнимая часть взаимного долеоктавного спектра.
- {Фаза} - фаза взаимного долеоктавного спектра.
- {Коэфф} - коэффициент когерентности взаимного долеоктавного спектра.
- {Ряд} - долеоктавный ряд.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- type ( 1/3 октавный) - тип анализа:
  1. 1/1 октавный.
  2. 1/3 октавный.

- 3. 1/12 октавный.
- 4. 1/24 октавный.
- Average Time (1) - время усреднения, с.
- spectrview (линейный) - тип представления уровня спектральных компонент:
  - 1. линейный масштаб (в единицах измерения)
  - 2. логарифмический масштаб (в децибелах)
- Activate (true) - состояние работы
- CalcRe (true) - расчёт действительной части спектра.
- CalcIm (true) - расчёт мнимой части спектра.
- CalcFase (true) - расчёт фазы спектра.
- CalcKoeff (true) - расчёт коэффициента когерентности спектра
- FileName - имя файла для записи результатов.

---

Иногда при статистическом анализе сигналов требуется накопление данных, а затем вычисление максимальных, минимальных, средних значений величин.

Примеры получения максимальных, минимальных, средних спектров, коррелограмм и т.д. представлены в проектах (на примере узкополосного спектра).



### ***Пример №1 вычисления максимальных величин:***

Данный пример непосредственно в ZETView

---



### ***Пример №2 вычисления минимальных величин:***

Данный пример непосредственно в ZETView

---



### ***Пример №3 вычисления средних величин:***

Данный пример непосредственно в ZETView



### 3.3. Взаимный корреляционный анализ

**Взаимный корреляционный анализ** - компонент. Предназначена для проведения взаимного корреляционного анализа сигналов, поступающих с входных каналов анализаторов спектра, для просмотра различных корреляционных характеристик сигналов.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Канал 1 - первый измерительный канал.
- Канал 2 - второй измерительный канал.
- Вкл/выкл - включение/выключение корреляционного анализа.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

##### ➤ Выходные

- Corr {Y} - массив значений корреляционной функции.
- {T} - массив значений задержек корреляционной функции.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке](#).<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

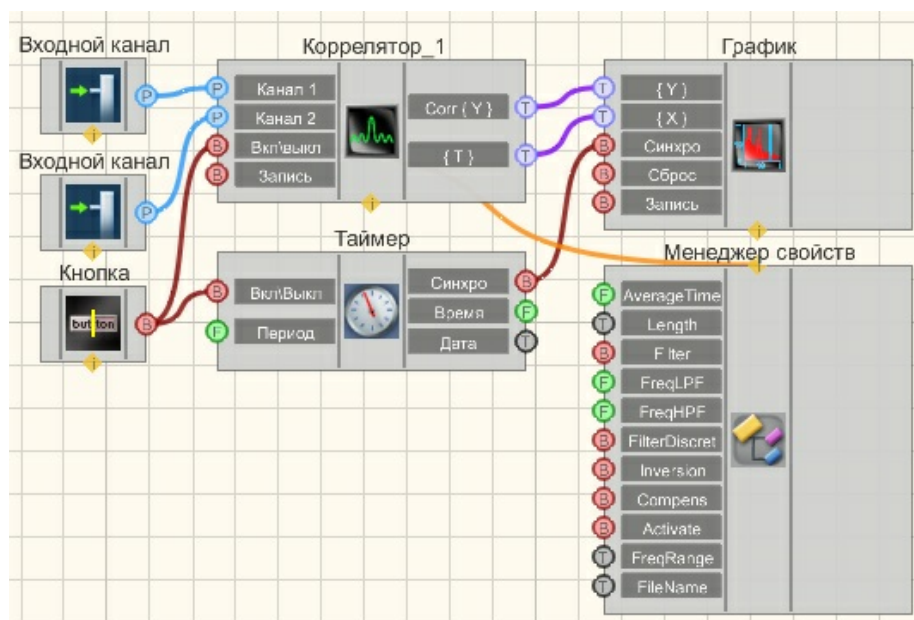
- AverageTime (1) - усреднение, с.
- length (128 точек) - размер корреляционной функции.
- Filter (false) - включение/выключение полосового фильтра.
- FreqLPF (10) - частоты среза фильтра низких частот, Гц.
- FreqHPF (1000) - частота среза фильтра высоких частот, Гц.
- FilterDiscret (false) - включение/выключение фильтрации дискретных помех.
- Inversion (false) - включение/выключение инверсии.
- Compens (false) - включение/выключение компенсации основного корреляционного пика.

- Activate (false) - состояние работы корреляционного анализа.
- freqrange (25000) - частотный диапазон, Гц
- FileName - имя файлов для записи результатов.

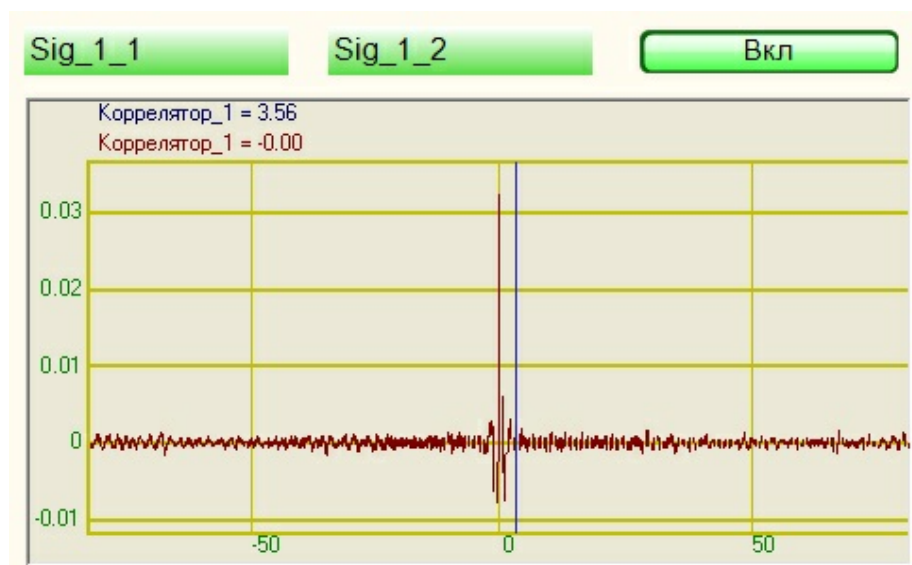


### Пример № 1

### Проект в SCADA ZETView



### Результат работы проекта



Данный пример непосредственно в ZETView

### 3.4. Взаимный узкополосный спектр

**Взаимный узкополосный спектр** - компонент. Предназначен для одновременного (взаимного) отображения нескольких узкополосных спектров. Мгновенный взаимный спектр рассчитывается комплексно сопряженным перемножением спектров двух каналов, полученных с помощью преобразования Фурье.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входные:

- Канал 1 - первый измерительный канал.
- Канал 2 - второй измерительный канал.
- Вкл/выкл - состояние работы взаимного узкополосного спектра.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

➤ Выходные

- {Спектр} - текущий мгновенный взаимный долеоктавный спектр.
- {Re} - действительная часть взаимного долеоктавного спектра.
- {Im} - мнимая часть взаимного долеоктавного спектра.
- {Фаза} - фаза взаимного долеоктавного спектра.
- {Коэфф} - коэффициент когерентности взаимного долеоктавного спектра.
- {Ряд} - долеоктавный ряд.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- freqrange (25000) - частотный диапазон, Гц.
- FreqResolution (1000) - количество полос.
- AverageTime (1) - время усреднения, с.
- usredtype (линейное) - тип режима накопления и усреднения:
  1. Линейное.
  2. Экспоненциальное.
- analysistype (дискретное ПФ) - тип анализа:
  1. Быстрое преобразование Фурье
  2. Дискретное преобразование Фурье.
- functiontype (Хана) - тип весовой функции:
  1. Прямоугольная.
  2. Хана
  3. Хэмминга.
  4. Блэкмана.
  5. Барлета.
  6. Блэкмана std.
- calsture (без обработки) - тип обработки сигнала:
  1. дифференцирование второго порядка
  2. дифференцирование первого порядка
  3. без обработки интегрирования и дифференцирования
  4. интегрирование первого порядка
  5. интегрирование второго порядка
- CalcReal (false) - расчёт действительной части спектра.
- CalcImage (false) - расчёт мнимой части спектра.
- CalcFase (false) - расчёт фазы спектра.

- CalcKoeff (false) - расчёт коэффициента когерентности спектра
- CalcPereh (false) - расчёт переходной характеристики.
- FileName - имя файлов для записи результатов
- Activate (false) - состояние работы.

Иногда при статистическом анализе сигналов требуется накопление данных, а затем вычисление максимальных, минимальных, средних значений величин.

Примеры получения максимальных, минимальных, средних спектров, коррелограмм и т.д. представлены в проектах (на примере узкополосного спектра).



### *Пример № 1 вычисления максимальных величин:*

Данный пример непосредственно в ZETView



### *Пример №2 вычисления минимальных величин:*

Данный пример непосредственно в ZETView



### *Пример №3 вычисления средних величин:*

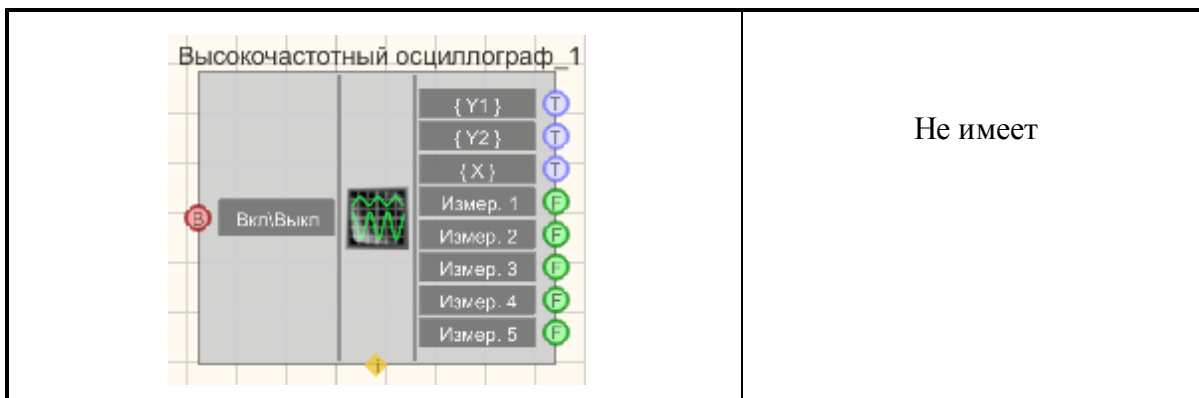
Данный пример непосредственно в ZETView

## 3.5.Высокочастотный осциллограф

**Высокочастотный осциллограф** - компонент. Предназначен для наблюдения высокочастотных сигналов с частотой до 20 МГц.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Вкл/Выкл - включение/выключение осциллографа.

#### ➤ Выходные

- {Y1} - выходной массив первого канала;
- {Y2} - выходной массив второго канала;
- {X} - размерность массива;
- Измер. 1 - значение первого измеренного параметра;
- Измер. 2 - значение второго измеренного параметра;
- Измер. 3 - значение третьего измеренного параметра;
- Измер. 4 - значение четвертого измеренного параметра;
- Измер. 5 - значение пятого измеренного параметра;

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Activate (true) - состояние работы высокочастотного осциллографа;
- FirstInput (true) - состояние работы первого канала;
- SecondInput (true) - состояние работы второго канала;
- firstresolution (5) - разрешение по первому каналу, В/клетка;
- secondresolution (5) - разрешение по второму каналу, В/клетка;
- firstinputtype (DC) - тип входа первого канала;
- secondinputtype (DC) - тип входа второго канала;
- firstinputprobe (1x) - пробник первого канала;
- secondinputprobe (1x) - пробник второго канала;
- FirstInversion (false) - инверсия данных первого канала;
- SecondInversion (false) - инверсия данных второго канала;
- synchroinput (Первый) - канал синхронизации;
- synchrofront (Восходящий) - фронт синхронизации:
  1. Восходящий.
  2. Нисходящий.


- synchromode (автоматический) - режим синхронизации:
  1. Автоматический.
  2. Обычный.
- SynchroLevel (0) - уровень синхронизации, клетки;
- timebase (1) - временная развёртка, с/клетка;
- Show (true) - разрешение/запрещение отображения окна осциллографа;
- measere1 (частота 1,Гц) - измеряемый параметр1:
  1. Частота 1, Гц;
  2. Частота 2, Гц;
  3. Период 1, с;
  4. Период 2, с;
  5. Среднее 1, В;
  6. Среднее 2, В;
  7. Размах 1, В;
  8. Размах 2, В;
  9. СКЗ 1, В;
  10. СКЗ 2, В;
  11. Минимальный уровень 1, В;
  12. Минимальный уровень 2, В;
  13. Максимальный уровень 1, В;
  14. Максимальный уровень 2, В;
  15. Нарастающий 1, с;
  16. Нарастающий 2, с;
  17. Спад 1, с;
  18. Спад 2, с;
  19. Положительный импульс 1, с;
  20. Положительный импульс 2, с;
  21. Отрицательный импульс 1, с;
  22. Отрицательный импульс 2, с;
- measere2 (частота 1,Гц) - измеряемый параметр2;
- measere3 (частота 1,Гц) - измеряемый параметр3;
- measere4 (частота 1,Гц) - измеряемый параметр4;
- measere5 (частота 1,Гц) - измеряемый параметр5.

---

### 3.6. Гистограмма

**Гистограмма** - компонент. Предназначен для гистографического анализа сигнала, нахождения статистических величин характеризующих сигнал и построения теоретических гистограмм по полученным данным.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - измерительный канал.
- Вкл/выкл - включение/выключение гистограммы.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

#### ➤ Выходные

- {Y} - рассчитанные значения по оси ординат.
- {Норм.} - рассчитанное нормальное распределение.
- {Гарм.} - рассчитанное гармоническое распределение.
- {Хи-квад.} - рассчитанное Хи-квадрат.
- {X} - разметка оси абсцисс.
- Мат.ожд. - математическое ожидание.
- Ст.откл. - стандартное отклонение.
- Дисперсия - выходной канал дисперсия.
- Ассиметрия - выходной канал асимметрии.
- Эксцесс - выходной канал эксцесс.
- Пик-фактор - выходной канал пик-фактор.
- Поле рас. - выходной канал поле рассеяния.
- Ц. рас. - выходной канал центр рассеяния.



- Откл. 1 - отклонение от нормального распределения.
- Откл. 2 - отклонение от гармонического распределения.
- Откл. 3 - отклонение от распределения Хи-квадрат.
- Энтропия - выходной канал энтропия.
- Коэфф. - энтропийный коэффициент качества.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- calcmode (по напряжению) - вариант расчета значений гистограммы:
  1. По напряжению.
  2. По разрядам.
- ColumnWidth (1) - ширина столбца гистограммы (в единицах измерения)
- calctype (гистограмма) - тип представления расчета:
  1. Гистограмма.
  2. Плотность вероятности.
  3. Вероятность.
- Interval (1) - время накопления данных, с.
- AverageTime (1) - время усреднения, с.
- CalcNormal (false) - расчет нормального распределения.
- CalcHarmonic (false) - расчет гармонического распределения.
- CalcHiSquare (false) - расчет распределения Хи-квадрат.
- HiSquareNumber (1) - число степеней свободы распределения Хи-квадрат
- Activate (true) - состояние работы.
- FileName - имя файла для записи результатов.

---

### 3.7. Детектор землетрясений

**Детектор землетрясений** - компонент предназначен для обнаружения и классификации сейсмических событий в радиусе около 200 км.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Канал X - имя канала компоненты X;
- Канал Y - имя канала компоненты Y;
- Канал Z - имя канала компоненты Z;
- Вкл\выкл - включение детектора землетрясений.

#### ➤ Выходные

- Да/нет - наблюдается или нет событие;
- Тип - тип последнего события;
- Магнитуда - магнитуда последнего события;
- P-фаза, tp - время вступления P-фазы последнего события;
- S-фаза, ts - время вступления S-фазы последнего события;
- Дистанция - расстояние до эпицентра последнего события, км;
- Пеленг - пеленг направления прихода волны P-фазы, град. от 0 до 360;
- Данные X - исходные данные последнего события по компоненте X;
- Данные Y - исходные данные последнего события по компоненте Y;
- Данные Z - исходные данные последнего события по компоненте Z.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Fmin (0.5) - нижняя частота полосового фильтра при детектировании, Гц;
- Fmax (10) - верхняя частота полосового фильтра при детектировании, Гц;
- ThresholdSTA (3) - порог детектора STA/LTA;
- ThresholdMagnitudo (3.5) - порог землетрясений по магнитуде;
- LevelWavelet (4) - уровень разложения при вейвлет-фильтрации сигнала;
- TimeSTA (10) - длительность короткого окна детектора STA/LTA, сек;
- TimeLTA (190) - длительность длинного окна детектора STA/LTA, сек;
- TypeWavelet (2) - материнский вейвлет при выполнении вейвлет- фильтрации

сигнала;

- Activate (false) - состояние работы;
- SpeedPhaseP (4.9) - скорость распространения P-фазы, км/с;
- SpeedPhaseS (3.2) - скорость распространения S-фазы, км/с;
- BearingY0 (0) - пеленг вектора Y используемого сейсмоприёмника, град. от 0 до 360;
- GroupWork (true) - возможность работы компонента в группе;
- FminForMag (0.2) - нижняя частота полосового фильтра при расчёте магнитуды, Гц;
- FmaxForMag (20) - верхняя частота полосового фильтра при расчёте магнитуды, Гц;
- PreHistory (2) - время предыстории сигнала без события, в размах от длительности STA (не менее 2, не более 10);
- TimeAverage (0.2)- время усреднения при использовании поляризационного анализа, в размах от длительности STA (не более 1);
- ThreshholdZigZag (0.4) - время значимости при использовании поляризационного анализа, от 0,1 до 1 сек;
- NeedDTU (false) - необходимость записи промежуточных результатов в dtu-файлы.

#### **Типы событий:**

- 1 - событие, у которого была определена только одна фаза;
- 2 - событие, у которого были определены две фазы, но максимальная амплитуда ускорения второй фазы меньше максимальной амплитуды ускорения первой фазы (не землетрясение);
- 3 - землетрясение, у которого магнитуда меньше заданного порога по магнитуде;
- 4 - землетрясение, у которого магнитуда больше заданного порога по магнитуде;
- -1 - событие, у которого при расчёте параметров получено некорректное значение.

#### **Вейвлет-фильтрация сигналов**

используется при детектировании первой фазы (отфильтровываются фоновые шумы, поэтому требуется задание времени предыстории сигнала без события) и при детектировании второй фазы (отфильтровываются сигналы первой фазы). Изменение типа материнского вейвлета и уровня разложения может привести к повышению загрузки процессора компьютера без улучшения точности определения времён вступления фаз землетрясения.

#### **Поляризационный анализ сигналов**

используется при определении времени действия первой фазы, а также при определении пеленга направления прихода волны P-фазы (не путать с пеленгом на эпицентр события!).

#### **Параметры TimeAverage и ThreshholdZigZag**

изменять не рекомендуется, т.к. это может привести к ухудшению точности определения времён вступления фаз землетрясения.

#### **На выходные ножки "P-фаза, tp" и "S-фаза, ts"**

выдаётся время вступления соответствующей фазы в текстовом виде в формате "ДД.ММ.ГГГГ ЧЧ:ММ:СС.ДС" (ДС - доли секунды), например: "13.05.2012 13:17:05.12" - 13 мая 2012 г. 13 часов, 17 минут, 5 секунд и 12 сотых секунды. Используется время компьютера (не UTC!!!). В случае работы с программой "Воспроизведение сигналов" время будет соответствовать времени записи сигналов.

#### **На выходные ножки "Данные ..."**

подаются массивы исходных данных по компонентам X, Y и Z. Эти данные не подвергались частотной фильтрации, и поэтому имеют постоянную составляющую выходного каскада используемого усилителя сигналов (или предварительного усилителя акселерометра). Время первого отсчёта этих массивов - это время вступления P-фазы, т.е. tp.

#### **Магнитуда землетрясения**

рассчитывается по известной формуле для определения магнитуды по объёмным волнам:

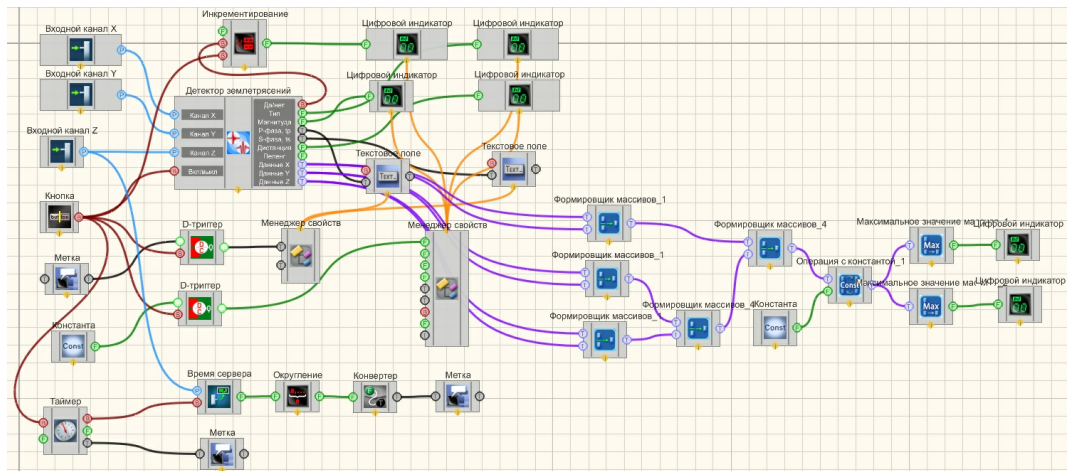
$$M = \lg \left( \frac{A_{\max}}{A_0} \right) + 2 \lg(2)$$

где  $A_{\max}$  - максимальное значение перемещения, зарегистрированное за время события,  $A_0 = 1$  мкм. Данная формула соответствует локальным землетрясениям в радиусе около 200 км.

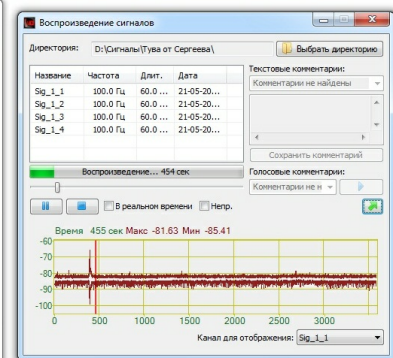
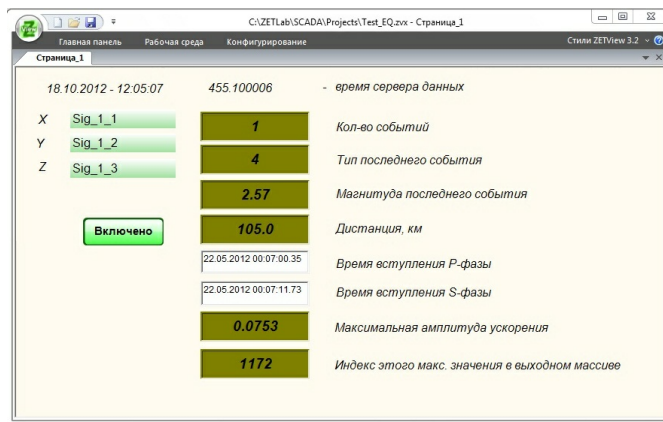


### ***Пример***

### ***Проект в SCADA ZETView***



## Результат работы проекта



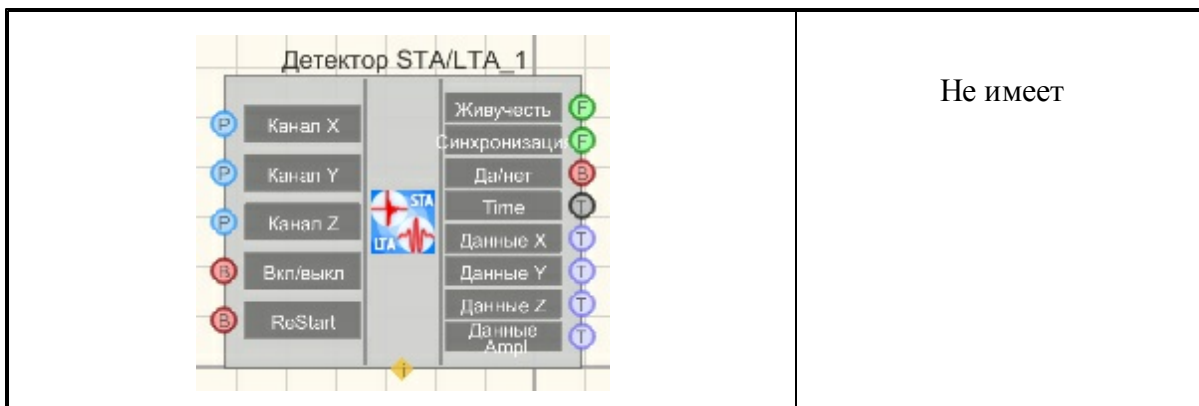
Данный пример непосредственно в ZETView

## 3.8. Детектор STA/LTA

**Детектор STA/LTA** - компонент «Детектор STA/LTA» (далее по тексту компонент) предназначен для детектирования сейсмических событий, регистрируемых одно- или трехкомпонентными сейсмоприёмниками.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Канал X - оцифрованные данные X сигнала сейсмоприёмника;
- Канал Y - оцифрованные данные Y сигнала сейсмоприёмника;
- Канал Z - оцифрованные данные Z сигнала сейсмоприёмника;
- Вкл\выкл - включение/выключение детектора STA/LTA;
- ReStart - команда на "горячий" перезапуск компонента.

#### ➤ Выходные

- Живучесть - сигнал живучести компонента.;
- Синхронизация - максимальное значение рассинхронизации сигналов X, Y, Z по времени;
- Да/нет - логический сигнал, наблюдается событие или нет;
- Time - абсолютное время (в текстовом виде) первого кадра данных, выдаваемых по X, Y, Z и по амплитуде сигнала;
- Данные X - данные события по X;
- Данные Y - данные события по Y;
- Данные Z - данные события по Z;
- Данные Ampl - данные события по амплитуде.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Activate (true) - режим работы компонента;
- signal (Векторный 3D) - тип сигнала: скалярный или векторный;
- PeriodVitality (1) - период, с которым компонент будет выдавать сигнал живучести, сек;
- Fmin (0,5) - частота среза ФВЧ при полосовой фильтрации сигнала, Гц;
- Fmax (10) - частота среза ФНЧ при полосовой фильтрации сигнала, Гц;
- TimeSTA - длительность короткого окна детектора STA/LTA, сек;
- TimeLTA - длительность длинного окна детектора STA/LTA, сек;
- TimeSTA(10) - длительность короткого окна детектора STA/LTA, сек;

- TimeLTA (190) - длительность длинного окна детектора STA/LTA, сек;
- Threshold (3) - порог срабатывания детектора STA/LTA;
- TimeBefore (5) - время до начала события, которое войдет в интервал выдаваемых данных события, может равняться 0, сек;
- TimeAfter (10) - время после начала события, которое войдет в интервал выдаваемых данных события, сек.

### Назначение

Компонент «Детектор STA/LTA» (далее по тексту компонент) предназначен для детектирования сейсмических событий, регистрируемых одно- или трехкомпонентными сейсмоприёмниками.

### Используемые компонентом методы ЦОС

Компонент выполняет полосовую фильтрацию выходных сигналов сейсмоприёмника, определяет абсолютное время начала события и выдаёт на выходные ножки массивы данных события по X, Y, Z и по амплитуде сигнала.

Фильтрация сигналов выполняется цифровыми фильтрами Баттерворта. Для детектирования событий используется одна из разновидностей широко используемых детекторов STA/LTA. Точное время начала события определяется с помощью вейвлет-фильтрации фоновых шумов сигналов, зарегистрированных непосредственно до события.

Свойствами компонента задаются длительности сигналов до и после начала события для выдачи массивов на выходные ножки.

### Работа компонента

При свойстве Activate = true компонент начинает свою работу сразу после запуска проекта. В противном случае работа начинается по приходу логической единицы на входную ножку *Вкл/Выкл*. Приход логического нуля на эту ножку в обоих случаях останавливает работу компонента.

Сразу после старта, а также после перезапуска во время инициализации детектора компонент выдаёт сигнал живучести, равный единице. Время инициализации детектора равно сумме TimeSTA и TimeLTA. После завершения инициализации детектора сигнал живучести обновляется компонентом с периодом PeriodVitality. В отсутствии события сигнал живучести компонента должен быть около единицы (но вероятность того, что новое значение будет в точности равняться старому значению очень мала), во время наблюдения события – больше единицы, сразу после события – меньше единицы, и даже может быть меньше 0.

После детектирования какого-либо события на ножку *"Да/нет"* сразу же будет выдана логическая единица. Через время, равное TimeAfter, компонент определит время начала события, которое в текстовом виде будет выдано на ножку *"Time"* в формате "ДД.ММ.ГГГГ ЧЧ:ММ:СС.хх", где хх – доли секунды. Также на выходные ножки, имена которых начинаются со слова *"Данные"* будут выданы массивы данных. Данные по X,

Y, Z и их амплитуда – это оцифрованные значения (и их амплитуда) исходных сигналов после полосовой фильтрации.

После завершения события (решение об этом компонент принимает самостоятельно) на ножку "Да/нет" будет выдан логический ноль.

Если величина сигнала живучести перестанет изменяться, то это может означать либо то, что компонент по какой-либо причине выполняет зациклившийся код, либо то, что исходные каналы X, Y, Z рассинхронизовались (например, при работе с сетевыми каналами из-за проблем с передачей данных по сети). Для диагностики проблем второго случая у компонента есть ножка "Синхронизация", на которую выводится максимальная разность времён каналов X, Y, Z, если это значение больше 1 сек. Небольшие значения, выдаваемые на эту ножку, не влияют на работу компонента. Однако может наступить такое состояние исходных сигналов, когда различие по их временам может привести к невозможности для компонента получать новые данные по этим каналам. В этом случае (когда сигнал живучести не меняется, а сигнал синхронизации больше 1) данные будут потеряны. При возобновлении нормального поступления исходных данных по каналам X, Y, Z компонент возобновит свою работу, и, как следствие, сигнал живучести начнёт меняться.

При зацикливании компонента (когда сигнал живучести не меняется, а сигнал синхронизации равен 0) необходимо на входную ножку "ReStart" кратковременно подать сигнал логической единицы, что приведёт к "горячему" перезапуску компонента. В случае рассинхронизации входных данных перезапуск компонента также может быть полезен.

### Замечания

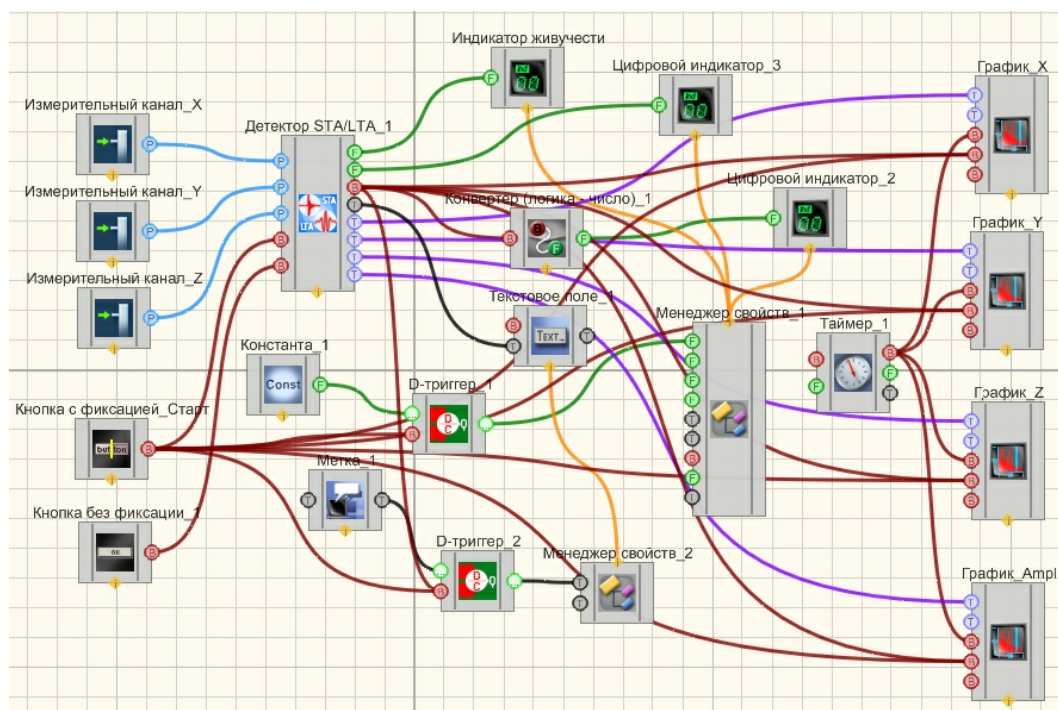
При определении максимальной величины сигнала в детектированном событии необходимо с запасом задавать свойство TimeAfter, т.к. возможна ситуация, когда максимальное значение будет наблюдаться позже начала события, чем время, равное TimeAfter. Например, при землетрясении, эпицентр которого находится на расстоянии от сейсмоприёмника около 200 км, S-фаза приходит после вступления P-фазы через время около 26 сек. Поэтому если установить TimeAfter и TimeSTA, равные 10 сек, то максимальное значение, которое всегда наблюдается во время S-фазы, может быть пропущено.



### Пример

### Проект в SCADA ZETView





*Результат работы проекта*

X

Y

Z

Вкл

*Амплитуда отфильтрованного сигнала*

Сигнал живучести 1.09618

Рассинхронизация каналов по времени 0.0

Событие (1 - наблюдается, 0 - нет) 0

Время первого кадра

*X-компонента отфильтрованного сигнала*

*Y-компонента отфильтрованного сигнала*

*Z-компонента отфильтрованного сигнала*

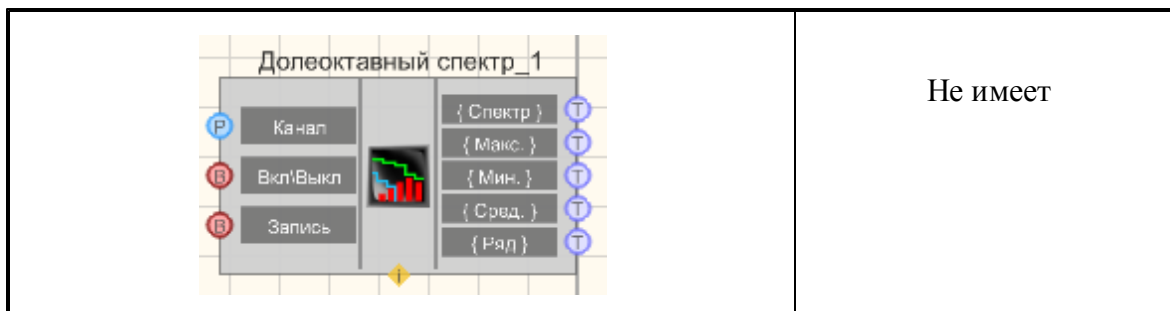
Данный пример непосредственно в ZETView

### 3.9. Долеоктавный спектр

**Долеоктавный спектр** - компонент. Предназначен для отображения различных параметров долеоктавных спектров.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Канал - измерительный канал.
- Вкл/выкл - включение/выключение долеоктавного спектра.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

#### ➤ Выходные

- {Спектр} - текущий мгновенный долеоктавный спектр.
- {Макс.} - текущий максимальный спектр.
- {Мин.} - текущий минимальный спектр.
- {Сред.} - текущий средний спектр.
- {Ряд} - долеоктавный ряд.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

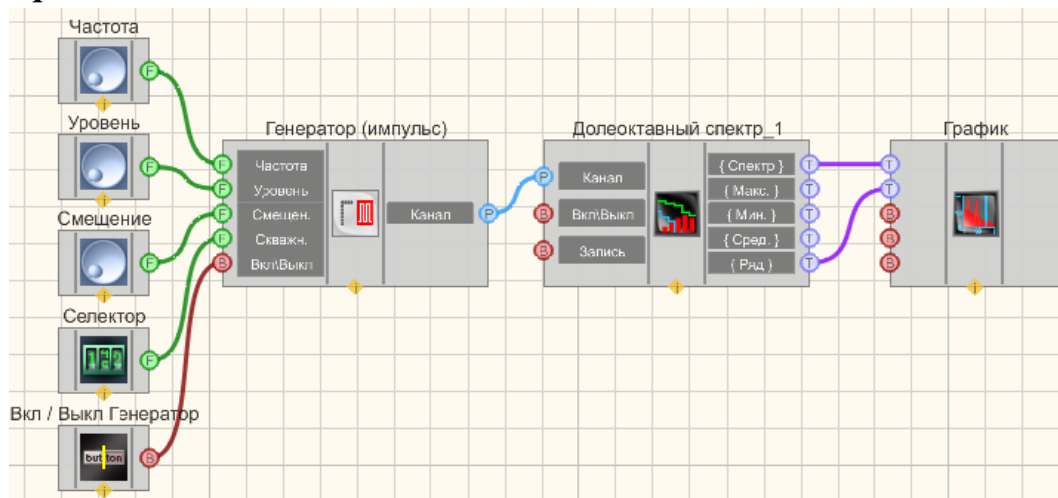
- type (1/3 октавный) - тип анализа:
  1. 1/1 октавный.
  2. 1/3 октавный.
  3. 1/12 октавный.
  4. 1/24 октавный.
- Average Time (1) - время усреднения, с.
- Activate (true) - состояние работы.
- calcType типа представления расчета спектра (СКЗ):
  1. среднеквадратичное значение
  2. пиковое значение
- spectrview (логарифмическое, дБ)-тип представления уровня спектральных компонент:
  1. линейный масштаб (в единицах измерения)
  2. логарифмический масштаб (в децибелах)
- CalcMax (false) - расчёт максимального спектра.
- CalcMin (false) - расчёт минимального спектра.

- CalcSred (false) - расчёт среднего спектра.
- CalcTime (100) - время расчёта дополнительных спектров, с
- FileName - имя файлов для записи результатов
- intdiff (без обработки)- выбор способа обработки данных (интегрирование, дифференцирование)



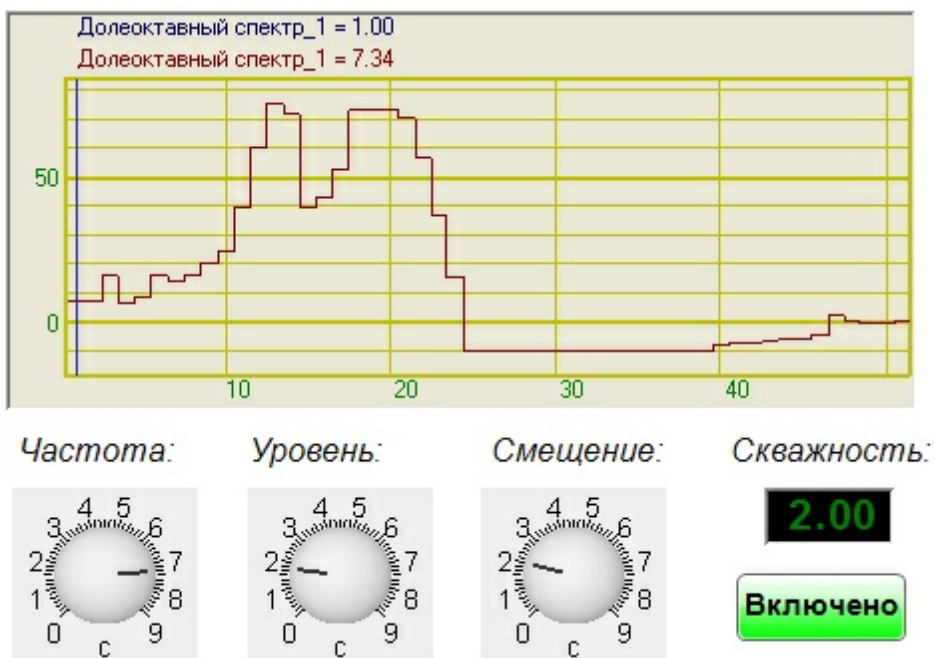
## Пример

### Проект в SCADA ZETView



В приведенном проекте компонент [Долеоктавный спектр](#)<sup>[120]</sup> измеряет среднеквадратичное значения пилообразного сигнала, поступающего с [Пилообразный сигнала](#)<sup>[239]</sup> генератора. [Селекторы](#)<sup>[213]</sup> ([также другого типа](#)<sup>[217]</sup>) служат для установки значений частоты, уровня, смещения и типа пилообразного сигнала. [График](#)<sup>[490]</sup> нужен для визуального восприятия сигнала. [ЖК индикатор](#)<sup>[348]</sup> служит для графического представления сигнала.

### Результат работы проекта



Данный пример непосредственно в ZETView

Иногда при статистическом анализе сигналов требуется накопление данных, а затем вычисление максимальных, минимальных, средних значений величин.

Примеры получения максимальных, минимальных, средних спектров, коррелограмм и т.д. представлены в проектах (на примере узкополосного спектра).



### ***Пример № 1 вычисления максимальных величин:***

Данный пример непосредственно в ZETView



### ***Пример №2 вычисления минимальных величин:***

Данный пример непосредственно в ZETView



### *Пример №3 вычисления средних величин:*

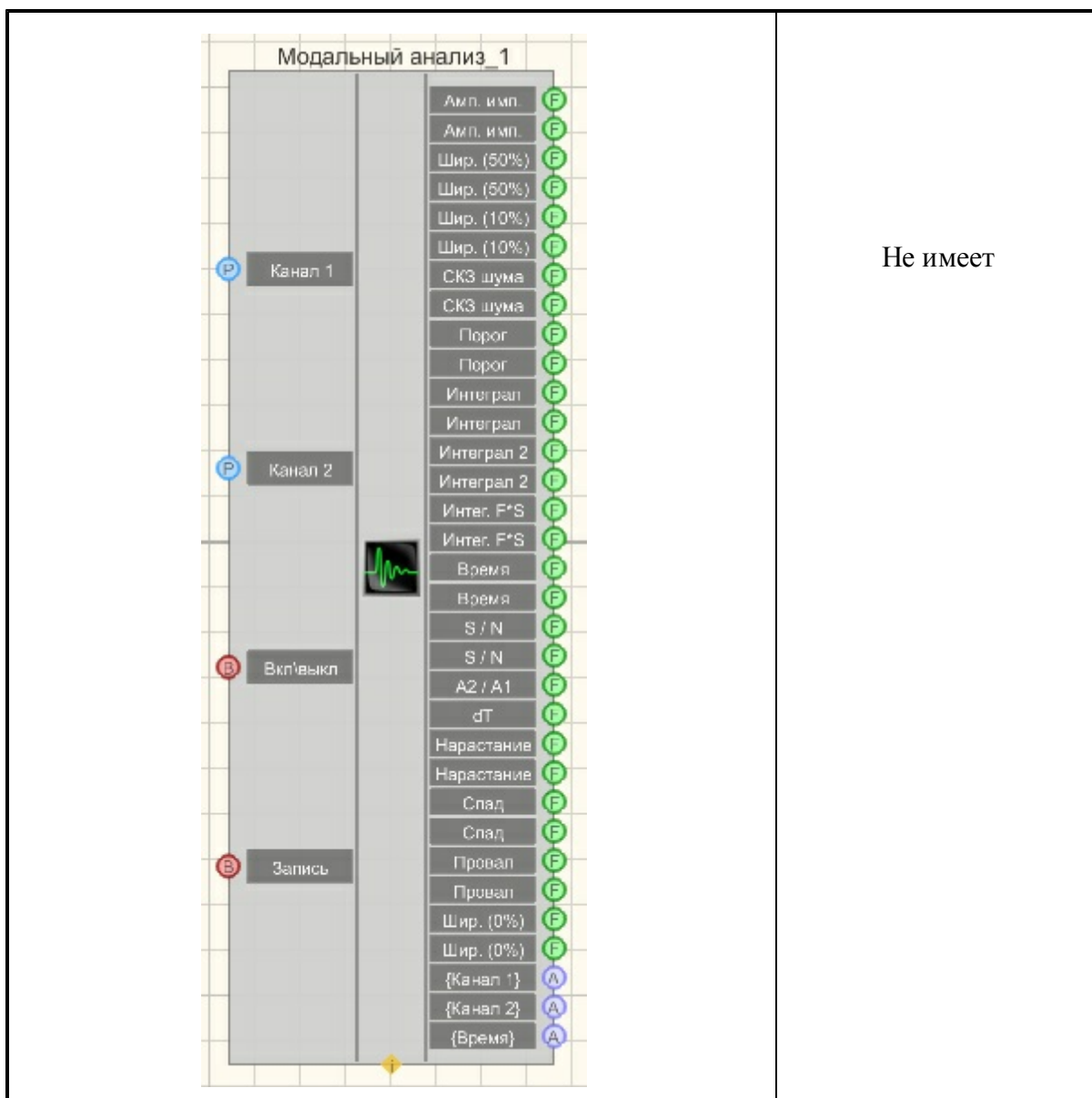
Данный пример непосредственно в ZETView

## 3.10.Модальный анализ

**Модальный анализ**- компонент. Предназначен для анализа импульсных и переходных характеристик сигналов, поступающих с входных каналов анализаторов спектра и сейсмостанций в реальном масштабе времени или в режиме воспроизведения записанных временных реализаций.

### **Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Канал 1 - опорный канал.
- Канал 2 - измерительный канал.
- Вкл\выкл - включение модального анализа.
- Запись - запись в файл

#### ➤ Выходные

- Ампл.имп. - амплитуда импульса на опорном канале.
- Ампл.имп. - амплитуда импульса на измерительном канале.
- Шир.(50%) - ширина импульса на опорном канале по 50-процентному уровню.
- Шир.(50%) - ширина импульса на измерительном канале по 50-процентному

уровню.

- Шир.(10%) - ширина импульса на опорном канале по 10-процентному уровню.
- Шир.(10%) - ширина импульса на измерительном канале по 10-процентному уровню.
- СКЗ Шума - СКЗ шума на опорном канале до импульса.
- СКЗ Шума - СКЗ шума на измерительном канале до импульса.
- Порог - порог срабатывания на импульс на опорном канале.
- Порог - порог срабатывания на импульс на измерительном канале.
- Интеграл - интеграл первого порядка от импульса на опорном канале.
- Интеграл - интеграл первого порядка от импульса на измерительном канале.
- Интеграл 2 - интеграл второго порядка от импульса на опорном канале.
- Интеграл 2 - интеграл второго порядка от импульса на измерительном канале.
- Интег.F\*S - интеграл произведения силы на перемещение на опорном канале.
- Интег.F\*S - интеграл произведения силы на перемещение на измерительном канале.
- Время - время прохождения импульса на опорном канале.
- Время - время прохождения импульса на измерительном канале.
- S/N - соотношение сигнал-шум на опорном канале
- S/N - соотношение сигнал-шум на измерительном канале.
- A2/A1 - отношение амплитуд импульсов на опорном и измерительном каналах.
- dT - разность времени следования импульсов на измерительном и опорном каналах.
- Нарастание - время нарастания сигнала на опорном канале.
- Нарастание - время нарастания сигнала на измерительном канале.
- Спад - время спада сигнала на опорном канале.
- Спад - время спада сигнала на измерительном канале.
- Провал - минимум сигнала перед ударом на опорном канале.
- Провал - минимум сигнала перед ударом на измерительном канале..
- Шир. (0%) - ширина импульса на опорном канале по 0-му уровню.
- Шир. (0%) - ширина импульса на измерительном канале по 10-ому уровню.
- {Канал1} - данные с опорного канала.
- {Канал 2} - данные с измерительного канала.
- {Время} - временной ряд.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- freqrange (50000)- частотный диапазон, Гц.
- ReferenceChanInversion (false) - инверсия опорного канала.
- MeasuringChanInversion (false) - инверсия измерительного канала.
- MeasuringChanLocalization (false) - нормировка измерительного канала.



- thresholdtype (Адаптивный (СКЗ\*К))- порог по СКЗ шумов (в единицах измерения).
- Coefficient (10) - множитель СКЗ адаптивного порога.
- AbsoluteReferenceChan (10) - абсолютный уровень СКЗ опорного канала (в единицах измерения).
- AbsoluteMeasuringChan (10) - абсолютный уровень СКЗ измерительного канала (в единицах измерения).
- fronttype (любой) - фронт запуска.
- QualityInterval (50) - интервал расчета добротностей, %
- Interval (1) - интервал расчёта, с.
- Auto (false) - установка автозапуска.
- AutoTime (10) - интервал автозапуска, с.
- Activate (true) - состояние работы.
- FileName - имя файлов для записи результатов

### 3.11.Обнаружитель событий

**Обнаружитель событий** - предназначен для обнаружения произвольных событий, сопровождаемых кратковременным увеличением амплитуды входного сигнала.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Канал X - X компонента сигнала;
- Канал Y - Y компонента сигнала;
- Канал Z - Z компонента сигнала;
- Вкл/выкл - включение обнаружителя событий.

##### ➤ Выходные

- Импульс - импульс об обнаружении события.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

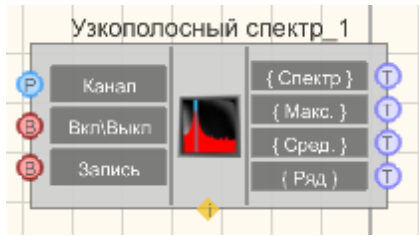
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- signaltype (скалярный) - исходный сигнал:
  1. скалярный;
  2. векторный 3D;
- LowFreq (1) - нижняя частота среза полосового фильтра, Гц;
- HighFreq (10) - верхняя частота среза полосового фильтра, Гц;
- STADuration (0.032720) - длительности короткого окна детектора STA/LTA, сек;
- LTADuration (0.32720) - длительности длинного окна детектора STA/LTA, сек;
- Thershold (false) - порог детектирования;
- NeedVirtChan (false) - создание виртуального канала с данными сигнала детектора STA/LTA, Да/Нет;
- NeedDTU (false) - запись данных, содержащих событие в dtu-файл, Да/Нет;
- NeedLog (false) - ведение log-файла компонента, Да/Нет;
- Active (false) - состояние компонента.

### 3.12. Узкополосный спектр

**Узкополосный спектр** - компонент. Предназначен для частотного анализа сигнала. По временной реализации сигнала находятся отклики по набору частотных фильтров. Центральные частоты фильтров равномерно распределены по оси частот.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Канал - измерительный канал.
- Вкл/выкл - включение/выключение долеоктавного спектра.
- Запись - при подаче на данный контакт импульса происходит сохранение

данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

➤ Выходные

- {Спектр} - текущий мгновенный узкополосный спектра.
- {Макс.} - текущий максимальный спектр.
- {Сред.} - текущий средний спектр.
- {Ряд} - частотный ряд.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Average Time (1) - время усреднения, с.
- calcType (Среднее квадратичное) - тип представления расчета спектра:
  1. Спектральная плотность.
  2. Спектральная мощность.
  3. Среднее квадратичное значение.
  4. Пиковое значение.
- analysisType (Дискретное ПФ) - тип обработки анализа:
  1. Быстрое преобразование Фурье
  2. Дискретное преобразование Фурье.
- functionType (Хэмминга) - тип весовой функции:
  1. Прямоугольная.
  2. Хана
  3. Хэмминга.
  4. Блэкмана.
  5. Барлета.
  6. Блэкмана std.
- scaleType (Логарифмический масштаб, дБ) - типа представления уровня спектральных компонент:
  1. Линейная (в единицах измерения).
  2. Логарифмический масштаб (дБ)
- Activate (true) - состояние работы.
- freqrange (25000) - частотный диапазон, Гц.
- freqresolution (50) - количество полос.
- CalcMax (false) - расчёт максимального спектра.
- CalcSred (false) - расчёт среднего спектра.
- CalcTime (100) - время расчёта дополнительных спектров, с
- intdiff (без обработки) - установка типа обработки сигнала:
  1. дифференцирование второго порядка.
  2. дифференцирование первого порядка.
  3. без обработки интегрирования и дифференцирования.
  4. интегрирование первого порядка.
  5. интегрирование второго порядка.

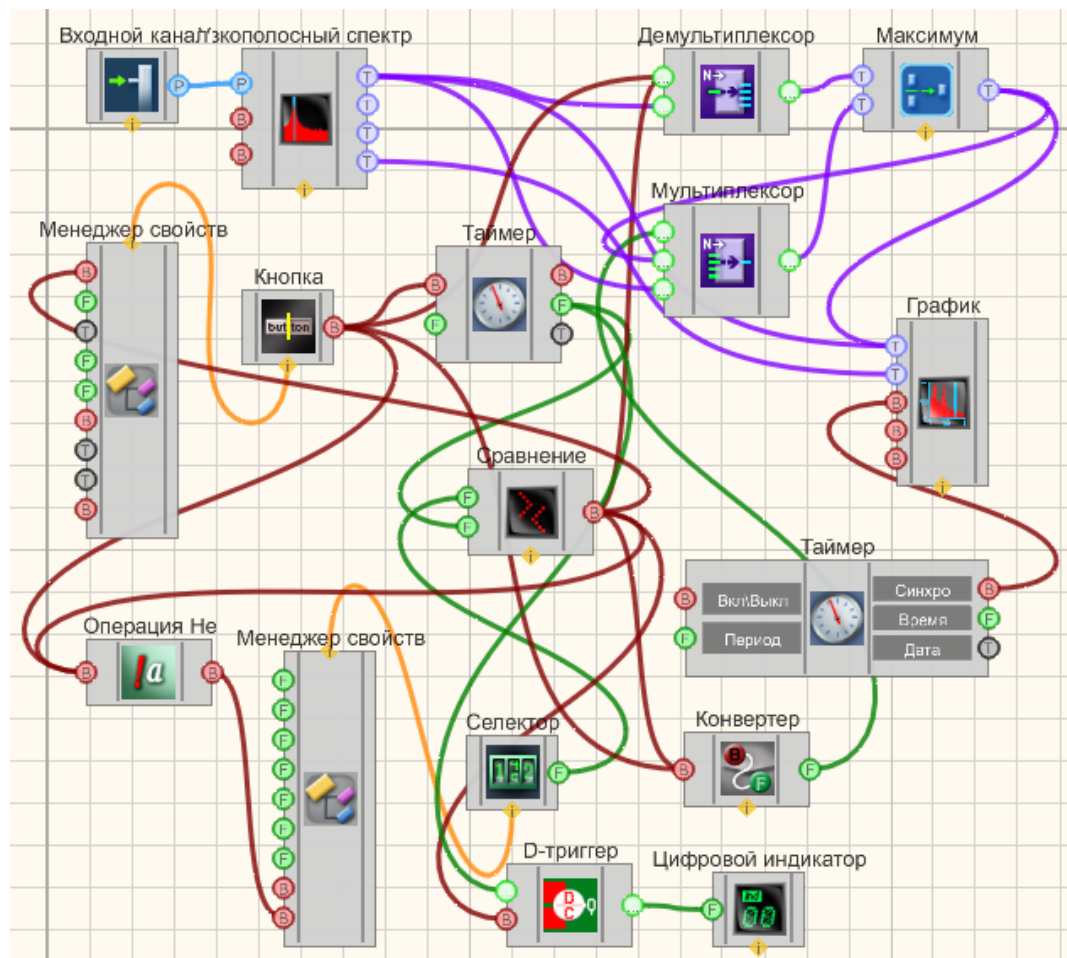
- Filter (false) - очистка спектра медианным фильтром.
- FileName - имя файлов для записи результатов.

Иногда при статистическом анализе сигналов требуется накопление данных, а затем вычисление максимальных, минимальных, средних значений величин. Примеры получения максимальных, минимальных, средних спектров, коррелограмм и т.д. представлены в проектах (на примере узкополосного спектра).

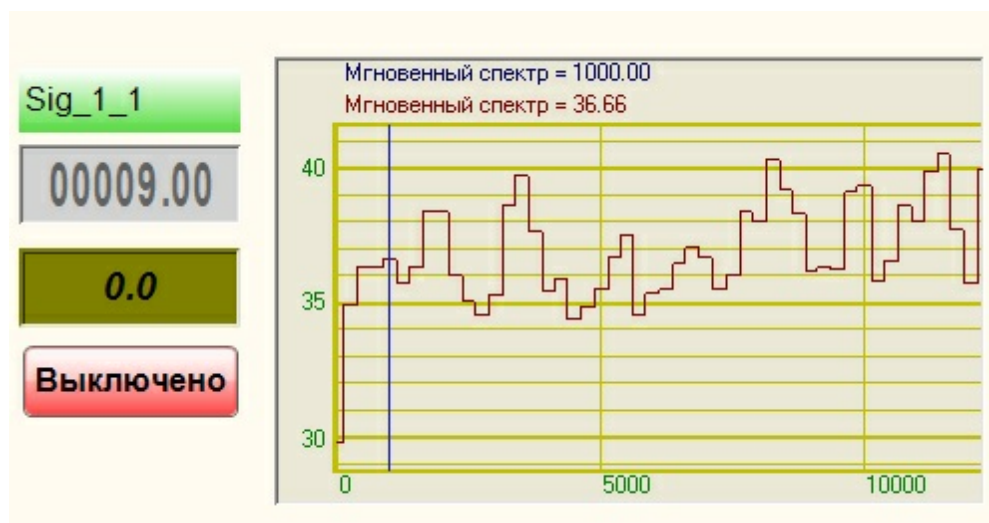


### *Пример № 1 вычисления максимальных величин:*

#### *Проект в SCADA ZETView*



#### *Результат работы проекта*



Данный пример непосредственно в ZETView



### *Пример №2 вычисления минимальных величин:*

Данный пример непосредственно в ZETView



### *Пример №3 вычисления средних величин:*

Данный пример непосредственно в ZETView

## Глава 4. Арифметика

### 4.1. Арифметика

**Арифметика** - универсальный арифметический компонент, принимающий два значения и производящий над ними одно из выбранных действий: сложение, вычитание, умножение, деление, степень, логарифм, корень, максимум, минимум, модуль.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - канал №1
- Вход 2 - канал №2
- Сброс - сбрасывает значение на входах 1 и 2

#### ➤ Выходные

- Выход - возвращает полученное значение, в зависимости от выбранного действия.

Назначение каналов компонента, в зависимости от выбранного действия:

Действие	Формула	Назначение канала №1 (A)	Назначение канала №2 (B)	Назначение выходного канала (C)
Сложение	$A+B=C$	Первое слагаемое	Второе слагаемое	Сумма
Вычитание	$A-B=C$	Уменьшаемое	Вычитаемое	Разность
Умножение	$A \cdot B=C$	Первый множитель	Второй множитель	Произведение
Деление	$A/B=C$	Делимое	Делитель	Частное
Степень	$A^B=C$	Основание	Показатель	Результат
Логарифм	$\text{Log}_{A_B}=C, B^C=A$	Число	Основание логарифма	Показатель степени
Корень	$\sqrt[B]{A}=C$	Подкоренное число	Степень корня	Результат
Максимум	$\text{Max}(A,B)$	Значение 1	Значение 2	Максимально

Действие	Формула	Назначение канала №1 (А)	Назначение канала №2 (В)	Назначение выходного канала (С)
				е значение
Минимум	$\text{Min}(A,B)$	Значение 1	Значение 2	Минимальное значение
Модуль	$ A,B  = \sqrt{A^2 + B^2}$	Значение 1	Значение 2	Модуль

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

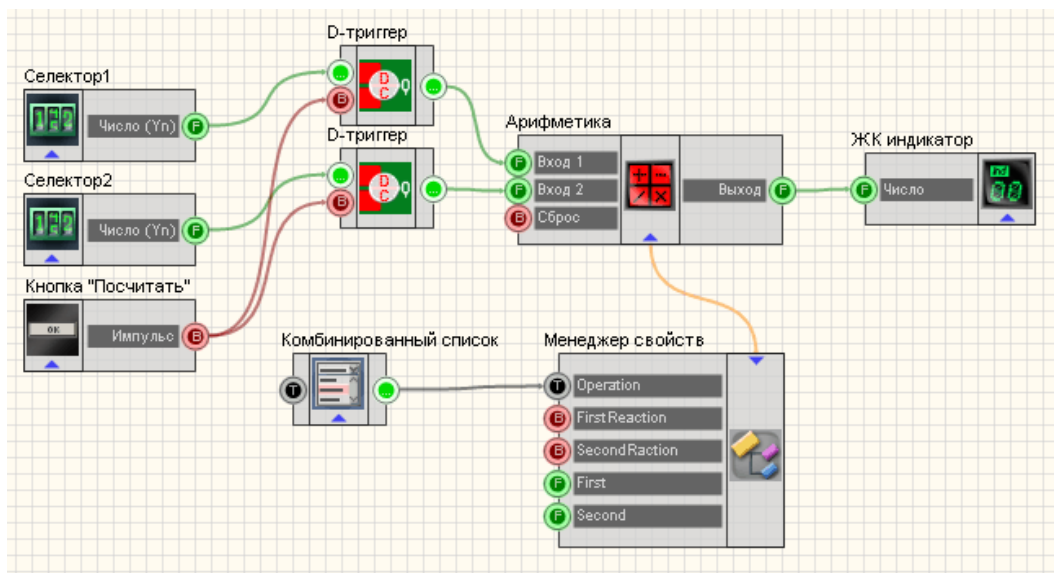
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Operation (Сложение)- Выбор вычислительной операции.
- First Reaction (true) - Реакция на изменение данных на входе №1:  
True - при изменении сигнала на входе изменится выходное значение.  
False - при изменении сигнала на входе выходное значение не изменится.
- Second Reaction (true) - Реакция на изменение данных на входе №2:  
True - при изменении сигнала на входе изменится выходное значение.  
False - при изменении сигнала на входе выходное значение не изменится.
- First (1) - Значение по умолчанию первого операнда.
- Second (1)
- - Значение по умолчанию второго операнда.



### *Пример*

### *Проект в SCADA ZETView*



В данном проекте рассматривается работа компонента "Арифметика". [Селекторы 1 и 2](#)<sup>[213]</sup> служат для того, чтобы выставить значения на первом и втором входах компонента "Арифметика". Значения с каждого из селекторов поступают на [D-триггеры](#)<sup>[384]</sup>, [кнопка без фиксации](#)<sup>[367]</sup> "Посчитать" посылает управляющий импульс на оба триггера, передавая выставленные на селекторах значения на первый и второй входы компонента "Арифметика". Так как "Арифметика" является многофункциональным компонентом, на нем можно выбрать действие, которое будет производиться с полученными значениями. Для этого к "Арифметике" на специальный вход подключается [Менеджер свойств](#)<sup>[534]</sup>, позволяющий вывести входы для дополнительных настроек компонента. Ко входу "Operation" на Менеджере свойств, отвечающему за выбор операции компонента "Арифметика", подключим компонент ["Комбинированный список"](#)<sup>[203]</sup> и внесем в его свойства названия всех действий, доступных для компонента "Арифметика". Далее Арифметика осуществляет выбранное действие над значениями со входов 1 и 2 и отправляет полученный результат на [цифровой индикатор](#)<sup>[348]</sup>.

### *Результат работы проекта*



Сначала выставляются значения на селекторах выбирается действие из комбинированного списка, затем нужно нажать кнопку "Посчитать", и тогда можно увидеть результат на цифровом индикаторе.



Посмотреть пример непосредственно в ZETView

## 4.2. Арктангенс

**Арктангенс** - компонент. Служит для реализации арифметической операции на выходе арктангенса операнда.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход X - на вход подается координата X.
- Вход Y - на вход подается координата Y.

#### ➤ Выходные

- Выход - выходной канал, на котором формируются арктангенс операнда.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

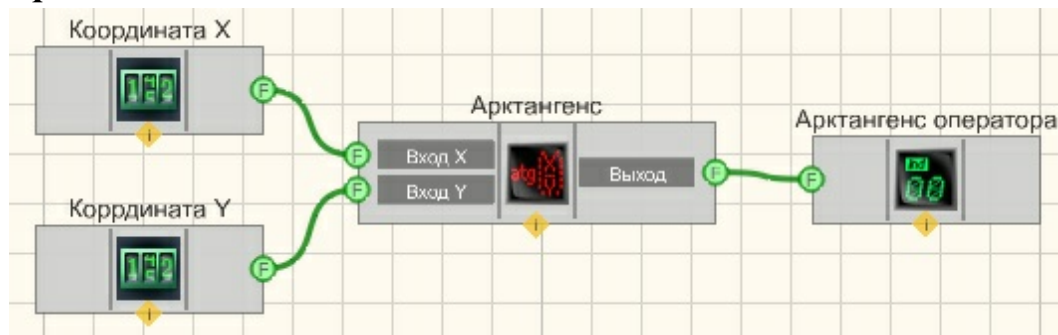
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе X:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе Y:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- First (0) - значение по умолчанию первого операнда (X).
- Second (0) - значение по умолчанию второго операнда (Y).



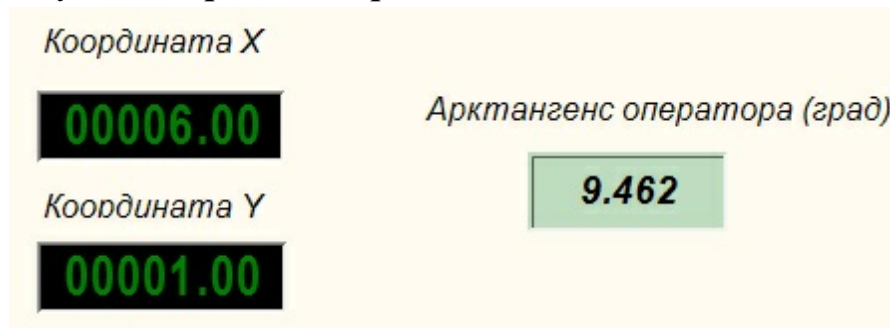
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Арктангенс](#)<sup>[135]</sup> производит операцию арктангенс двух чисел. [Селекторы](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента [Арктангенс](#)<sup>[135]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода арктангенса операнда.

### Результат работы проекта




Данный пример непосредственно в ZETView

## 4.3.Вычитание

**Вычитание** - компонент. Служит для реализации арифметической операции вычитания.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Вход 1 - канал №1 является уменьшаемым.
- Вход 2 - канал №2 является вычитаемым.

## ➤ Выходные

- Выход - выходной канал, на котором формируются разность численных значений каналов.

**Настраиваемые свойства:**

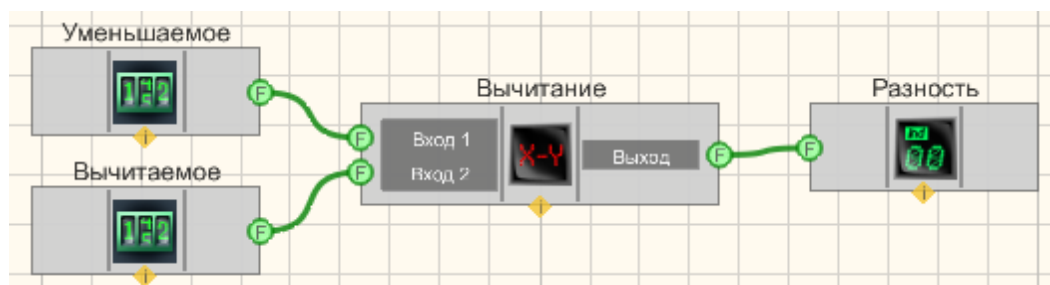
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

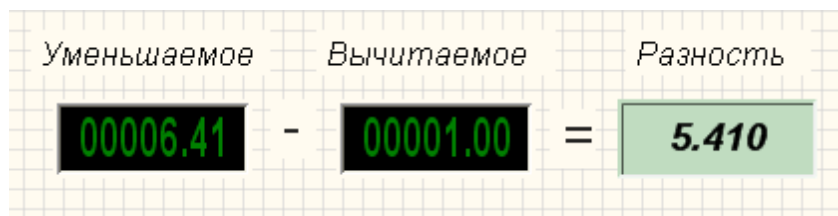
- FirstReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.

**Пример****Проект в SCADA ZETView**



В этой схеме компонент [Вычитание](#)<sup>[135]</sup> производит операцию разности двух чисел. [Селекторы](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента [Вычитание](#)<sup>[135]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода разности.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 4.4. Деление

**Деление** - компонент. Служит для реализации такой арифметической операции, как деление.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - значение на данном входе является делимым.

- Вход 2 - значение на данном входе является делителем.
- Выходные
- Выход - выходной канал, на котором формируются частное численных значений входов.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

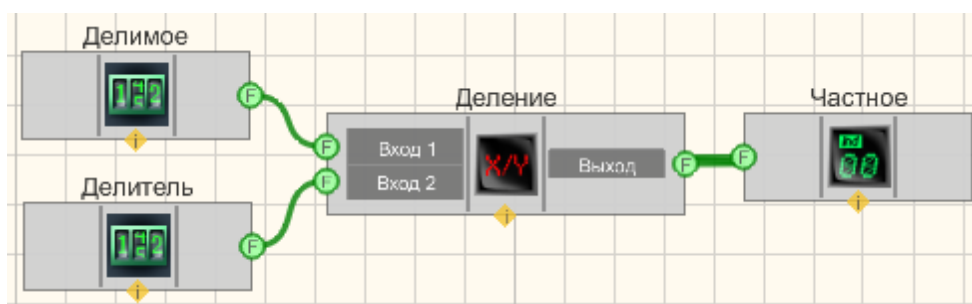
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true)- установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (true)- установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.



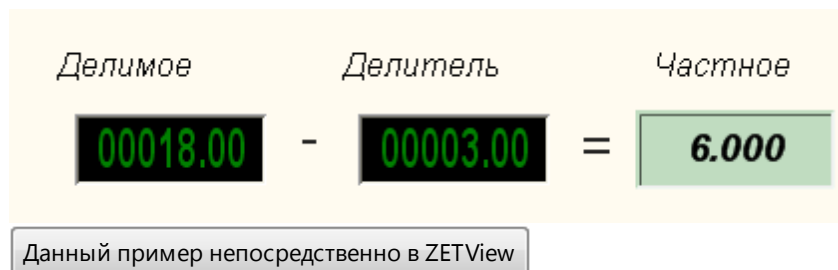
## Пример

### Проект в SCADA ZETView



В этой схеме компонент **Деление** производит операцию деления двух чисел. [Селекторы](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента **Деление**. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода частного.

## Результат работы проекта



## 4.5. Инкрементирование

**Инкрементирование** - компонент. Служит для реализации такой арифметической операции, как прибавление к численному значению входного сигнала константы.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной канал компонента.
- Импульс - входной контакт, при изменении на котором логического уровня, к численному значению входного канала прибавляется константа.
- Сброс - входной контакт, значение поступающее на вход компонента, при изменении логического уровня, сбрасывается в нуль.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, подвергшийся инкрементации.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (0) - текущее значение.
- Step (1) - шаг инкрементирования.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Инкрементирование](#)<sup>[140]</sup> постоянно прибавляет единицу к численному значению входного канала. [Таймер](#)<sup>[183]</sup> служит для генерации импульсов на вход [Инкрементирования](#)<sup>[140]</sup>, по факту которых производится инкремент входного значения. [Селекторы](#)<sup>[213]</sup> служат для установки входного значения и периода таймера. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического представления численного значения выходного сигнала.

### Результат работы проекта

<p><b>Настройка таймера:</b></p> <p>Период таймера: <span style="border: 1px solid black; padding: 2px;">00000,50</span></p> <p>Сброс таймера: <span style="border: 1px solid black; padding: 2px;">Сброс</span></p>	<p><b>Настройка компонента "Инкрементирование":</b></p> <p>Входное значение: <span style="border: 1px solid black; padding: 2px;">00011,00</span></p> <p>Результат инкрементирования: <span style="border: 1px solid black; padding: 2px;">26.0</span></p>
--	--

Данный пример непосредственно в ZETView

## 4.6. Интегратор

**Интегратор** - компонент. Служит для реализации такой математической операции, как интегрирование.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной канал компонента.
- Сброс - входной контакт, при изменении на котором логического уровня, сбрасывается в 0 значение, поступающее на вход компонента.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющий собой сумму всех предыдущих значений и текущего.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства:

Не имеет.



## Пример

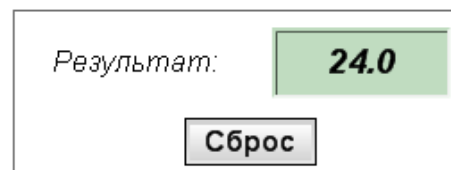
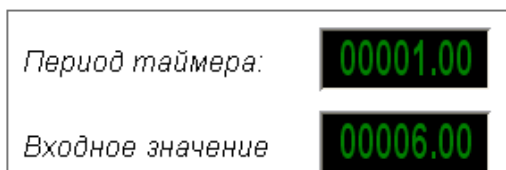
### Проект в SCADA ZETView



В этой схеме компонент [Интегратор](#)<sup>[141]</sup> постоянно суммирует входное значение, которое задается [Селектором](#)<sup>[213]</sup>. [Таймер](#)<sup>[183]</sup> подает синхроимпульсы на вход [D-триггера](#)<sup>[384]</sup>, чтобы обеспечить периодичность поступления численных значений в [Интегратор](#)<sup>[141]</sup>. [Селектор](#)<sup>[213]</sup> таймера позволяет задать периодичность этих синхроимпульсов. [Кнопка Сброс](#)<sup>[367]</sup> позволяет вернуть выходное значение [Интегратора](#)<sup>[141]</sup> к начальному значению. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления информации.



## Результат работы проекта



Данный пример непосредственно в ZETView

## 4.7. Логарифм

**Логарифм** - компонент. Служит для реализации такой математической операции, как логарифмирование.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - входной канал компонента, на который подается основание логарифма.
- Вход 2 - входной контакт, число под логарифмом (число при логарифмировании которого по основанию (Вход 1) получится выходное значение).

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющийся показателем степени, в которую надо возвести основание (Вход 1), чтобы получить число на Входе 2.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - установка реакции изменения выходного сигнала на

изменение значений на входе №1:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

значение.

- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

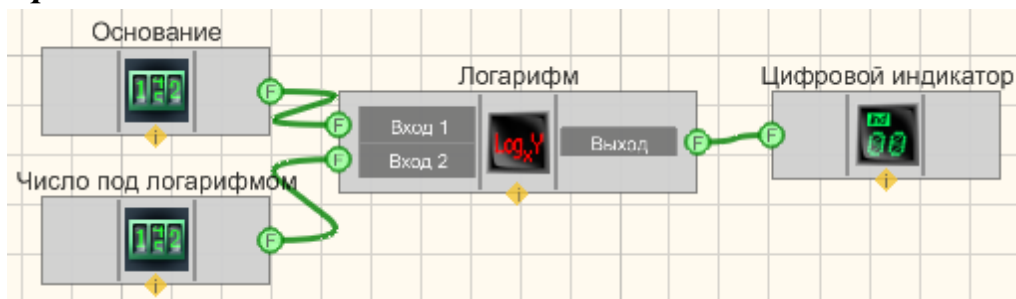
значение.

- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.
- value (0) - выходное значение.



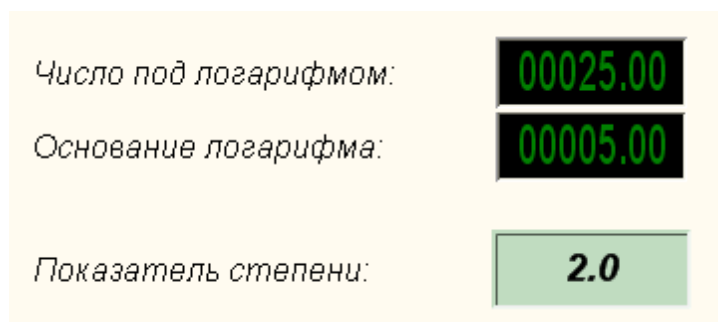
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Логарифм](#)<sup>[143]</sup> служит для нахождения значения логарифма от значения, поступающего с [Селектора](#)<sup>[213]</sup> (Число под логарифмом) на Вход-2 по основанию, задаваемому [Селектором](#)<sup>[213]</sup> (Основание) на Вход-1. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического представления численного значения выходного сигнала.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 4.8. Модуль

**Модуль** - компонент. Служит для реализации такой арифметической операции, как получение абсолютного значения величины.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - контакт, на который поступают, численные значения.

#### ➤ Выходные

- Выход - выходной канал, на котором формируются канал абсолютных значений.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

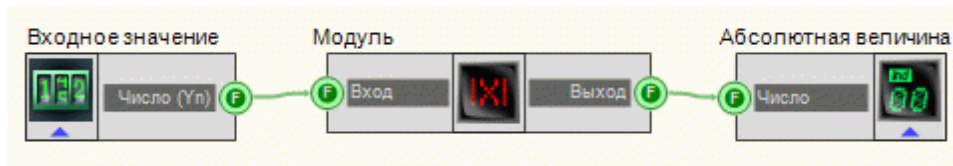
#### ➤ Частные свойства:

Не имеет.



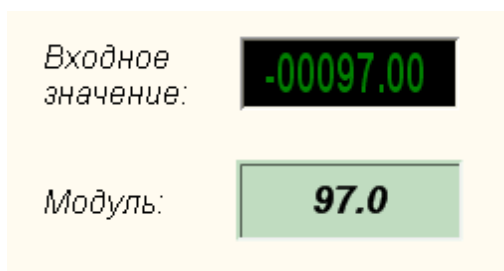
*Пример*

## Проект в SCADA ZETView



В этой схеме компонент [Модуль](#)<sup>[145]</sup> производит операцию получения абсолютного значения числовых данных в канале. [Селектор](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента [Модуль](#)<sup>[145]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода абсолютных значений.

## Результат работы проекта



Данный пример непосредственно в ZETView

## 4.9. Округление

**Округление** - компонент. Служит для реализации такой арифметической операции, как округление.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - контакт, на который поступают, численные значения.

#### ➤ Выходные

- Выход - выходной канал, на котором формируются канал округленных значений.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

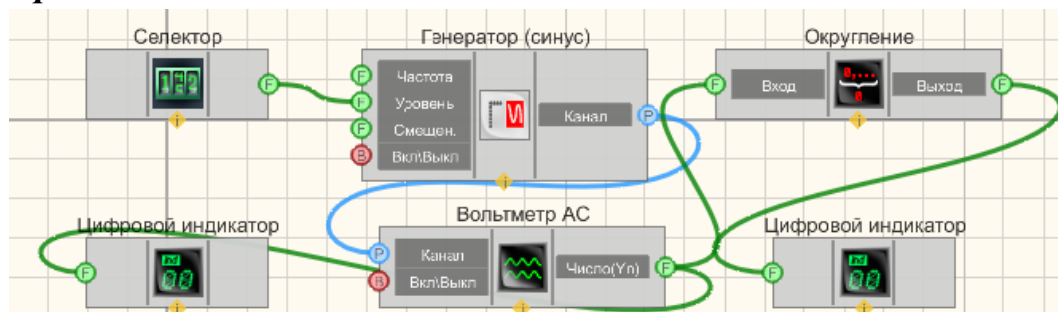
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Precision (0) - количество знаков после запятой (точность округления).

**Пример**

*Данный пример требует подключенного ZET-устройства, содержащего ЦАП.*

**Проект в SCADA ZETView**

[Селектор](#)<sup>[213]</sup> служит для ввода значения амплитуды синусоидального сигнала. [Вольтметр АС](#)<sup>[284]</sup> производит измерение среднеквадратического значения сигнала. В этой схеме компонент [Округление](#)<sup>[146]</sup> производит операцию округления до одного знака после занятой. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода уже округленного значения. [Генератор \(синус\)](#)<sup>[247]</sup> нужен для получения гармонического сигнала. Второй [Цифровой индикатор](#)<sup>[362]</sup> нужен для отображения значения напряжения, снятого непосредственно с вольтметра.

**Результат работы проекта**



## 4.10.Остаток от деления

**Остаток от деления** - компонент. Служит для реализации такой арифметической операции, как получения остатка от деления.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - контакт, на который поступает численное значение, являющееся делимым.
- Вход 2 - контакт, на который поступает численное значение, являющееся делителем.

#### ➤ Выходные

- Выход - выходной канал, на котором формируется остаток от деления численных значений канала №1 на канал №2.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - установка реакции изменения выходного сигнала на

изменение значений на входе №1:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

значение.

- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

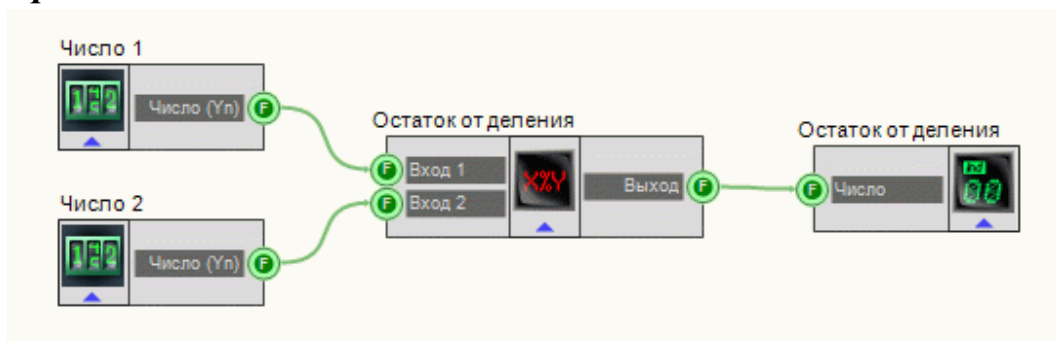
значение.

- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.



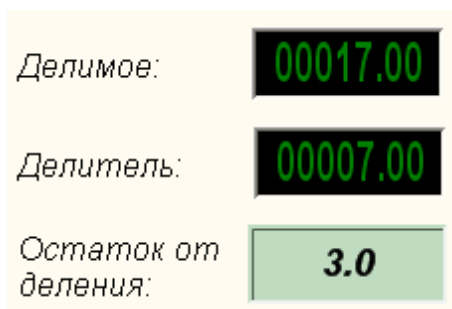
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Остаток от деления](#)<sup>[148]</sup> позволяет нам получить остаток от деления числа 1 на число 2. [Селектор](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента [Остаток от деления](#)<sup>[148]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода уже округленного значения.

### Результат работы проекта

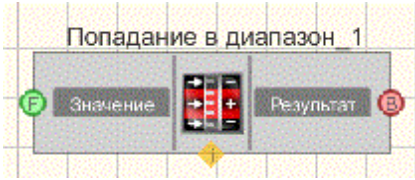


Данный пример непосредственно в ZETView

## 4.11. Попадание в диапазон

**Попадание в диапазон** - компонент. Позволяет выставить верхнюю и нижнюю границы диапазона, проверяя, входит ли полученное на входе значение в указанный диапазон. На выход подает 1- в случае, когда значение попадает в диапазон и 0 - в противном случае.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Значение - Входное значение

#### ➤ Выходные

- Результат - сигнальный выход, с которого подается значение "1" в случае, когда входное число попадает в указанный диапазон, "0" - в случае, когда входное число не попадает в указанный диапазон.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

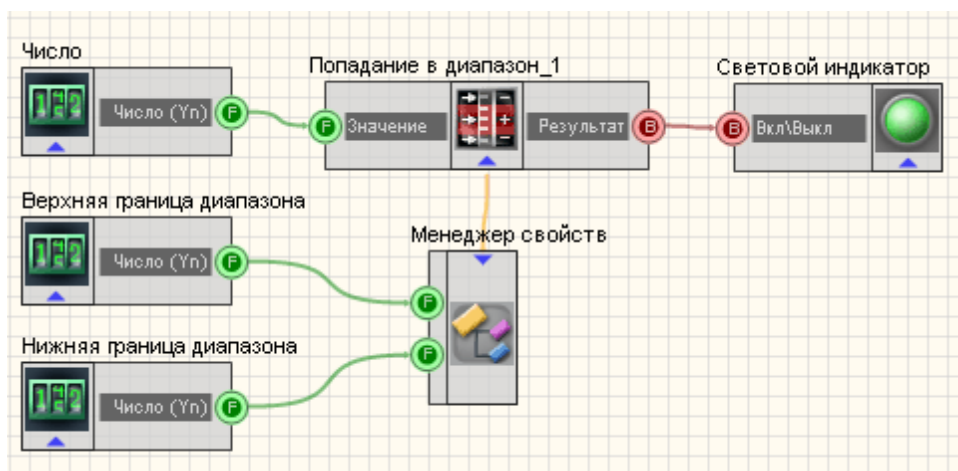
- fRangeStart (0) - нижняя граница диапазона.
- fRangeEnd (0) - верхняя граница диапазона.



### *Пример*

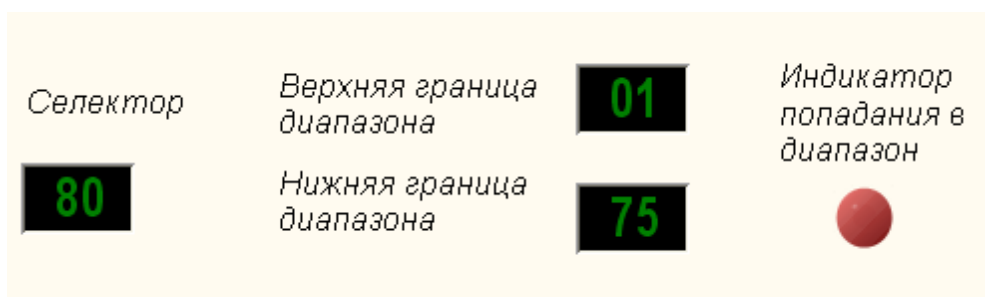
### *Проект в SCADA ZETView*





В данном проекте рассматривается работа компонента "[Попадание в диапазон](#)"<sup>[150]</sup>. [Селекторы](#)<sup>[213]</sup> служат для того, чтобы выставить входное значение для компонента "[Попадание в диапазон](#)"<sup>[150]</sup>, а также значения верхней и нижней границ диапазона. [Менеджер свойств](#)<sup>[534]</sup>, позволяет вывести дополнительные входы для настройки свойств компонента "[Попадание в диапазон](#)"<sup>[150]</sup>. Результат подается на вход компонента "[Световой индикатор](#)"<sup>[356]</sup>. В случае попадания в диапазон он загорится зеленым, в обратном случае - красным.

### Результат работы проекта




Посмотреть пример непосредственно в ZETView

## 4.12.Разность

**Разность** - компонент. Служит для реализации такой математической операции, как нахождение разности двух значений на одном входе.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной канал компонента.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющий собой разность текущего и предыдущего значений на входе.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства:

Не имеет.



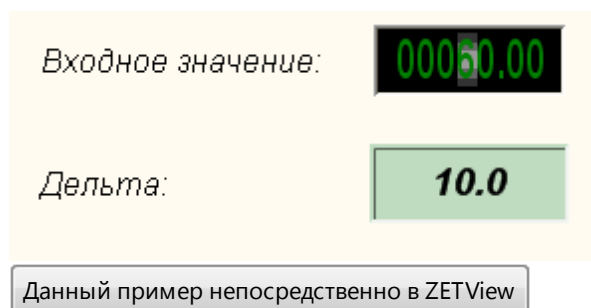
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Разность](#)<sup>[151]</sup> служит для нахождения разности текущего и предыдущего значений. [Селектор](#)<sup>[213]</sup> нужен для задания входных значений. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления выходных значений.

### Результат работы проекта



## 4.13.Сложение

**Сложение** - компонент. Служит для реализации такой математической операции, как сумма двух численных значений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - входной канал слагаемого №1.
- Вход 2 - входной канал слагаемого №2.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющий собой сумму двух входных численных значений.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.

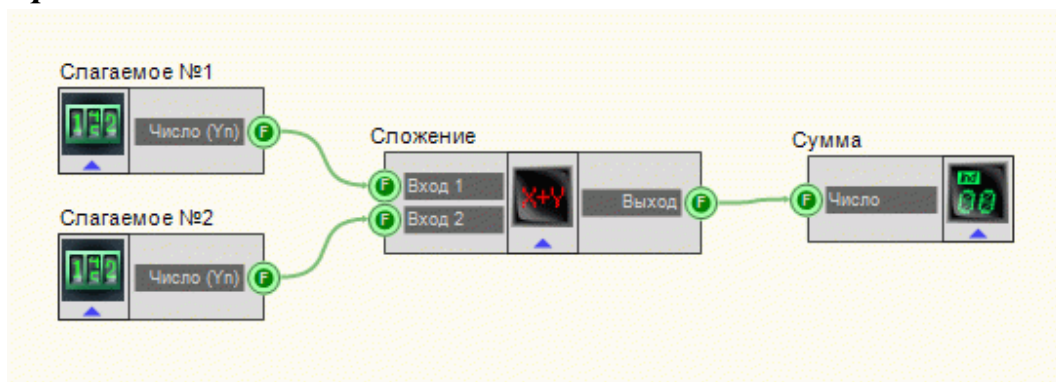
2. False - при изменении сигнала на входе - не изменится выходное значение.

- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент "[Сложение](#)"<sup>[153]</sup> служит для нахождения суммы двух входных значений. [Селекторы](#)<sup>[213]</sup> нужны для выставления численных значений слагаемым. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления выходных значений.


### Результат работы проекта

Данный пример непосредственно в ZETView

## 4.14.Сравнение (неравенство)

**Сравнение (неравенство)** - компонент. Служит для реализации такой математической операции, как сравнение двух численных значений.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Вход 1 - входной канал сравнения №1.
- Вход 2 - входной канал сравнения №2.

## ➤ Выходные

- Выход - выходной контакт, на котором формируется высокий логический уровень в случае, если значение на входе №1 больше или равно значения на входе №2, в противном же случае - низкий.

**Настраиваемые свойства:**

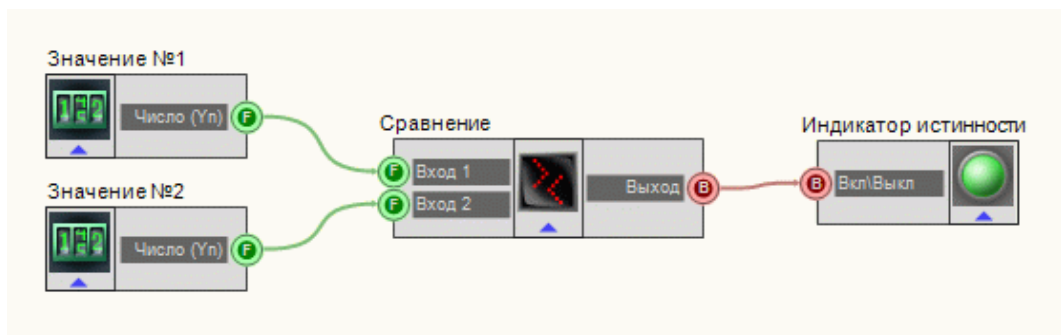
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию)

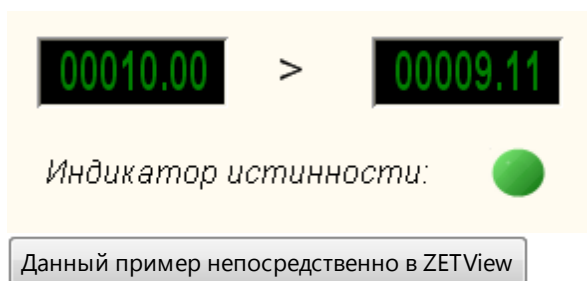
- FirstReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.

**Пример****Проект в SCADA ZETView**



В этой схеме компонент [Сравнение](#)<sup>[154]</sup> служит для сопоставления двух входных значений между собой. [Селекторы](#)<sup>[213]</sup> нужны для выставления численных значений сравниваемым величинам. [Световой индикатор](#)<sup>[356]</sup> служит для визуального представления результата сравнения. В случае, если значение 1 больше или равно значению 2, то индикатор загорается зеленым цветом, иначе - красным.

### Результат работы проекта



## 4.15.Сравнение (равенство)

**Сравнение (равенство)** - компонент. Служит для реализации такой математической операции, как проверка на равенство двух численных значений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

➤ Входы:

- Вход 1 - входной канал сравнения №1.
- Вход 2 - входной канал сравнения №2.

➤ Выходные

- Выход - выходной контакт, на котором формируется высокий логический уровень в случае, если значение на входе №1 равно значению на входе №2, в противном же случае - низкий.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №1:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное значение.

- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.



## Пример

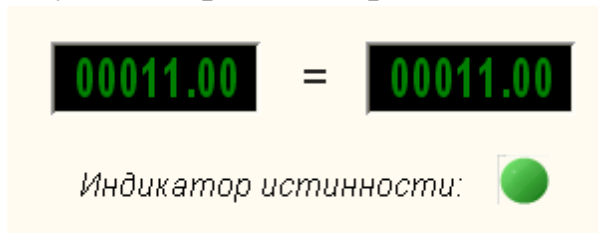
### Проект в SCADA ZETView



В этой схеме компонент [Сравнение](#)<sup>[156]</sup> служит для сопоставления двух входных значений между собой. [Селекторы](#)<sup>[213]</sup> нужны для выставления численных значений сравниваемым величинам. [Световой индикатор](#)<sup>[356]</sup> служит для визуального представления результата сравнения.

В случае равенства входных значений, индикатор загорается зеленым цветом, иначе - красным.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 4.16.Степень

**Степень** - компонент. Служит для реализации такой математической операции, как возведение числа в степень.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Число - входной канал компонента, на который подается число, которое требуется возвести в степень.
- Степень - входной канал компонента, на который подается значение возводимой степени.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющий собой значение числа, уже возведенного в степень.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке](#).<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true)- установка реакции изменения выходного сигнала на изменение значений на входе №1:

1. True - при изменении сигнала на входе - изменится выходное значение.
2. False - при изменении сигнала на входе - не изменится выходное



значение.

- SecondReaction (true) - установка реакции изменения выходного сигнала на изменение значений на входе №2:

1. True - при изменении сигнала на входе - изменится выходное значение.

2. False - при изменении сигнала на входе - не изменится выходное значение.

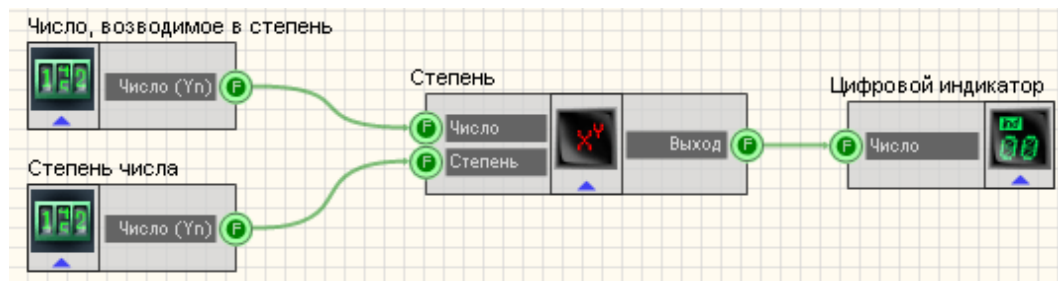
значение.

- First (1) - значение по умолчанию первого операнда.
- Second (1) - значение по умолчанию второго операнда.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Степень](#)<sup>[158]</sup> служит для нахождения значения числа после возведения его в степень. [Селекторы](#)<sup>[213]</sup> нужны для ввода числа и требуемой степени. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления выходных значений.

### Результат работы проекта

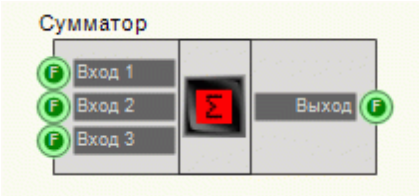
Число, возводимое в степень:	00007.00
Степень числа:	00002.00
Результат:	49.0

Данный пример непосредственно в ZETView

## 4.17.Сумматор

**Сумматор** - компонент. Служит для реализации такой математической операции, как единовременное суммирование входящих значений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - входной канал №1.
- ...
- Вход N - входной канал №N.

#### ➤ Выходные

- Выход - выходной контакт, на котором формируется канал, являющий собой сумму всех предыдущих значений и текущего.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

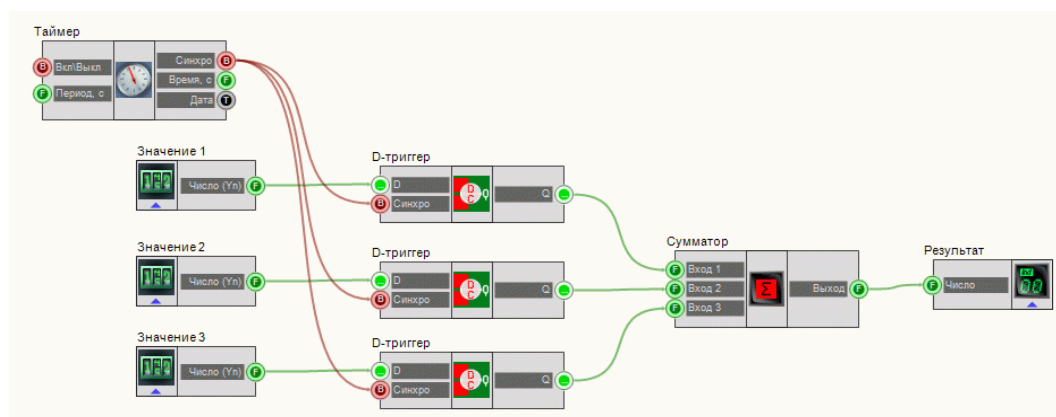
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Input (1) - установка числа входов.



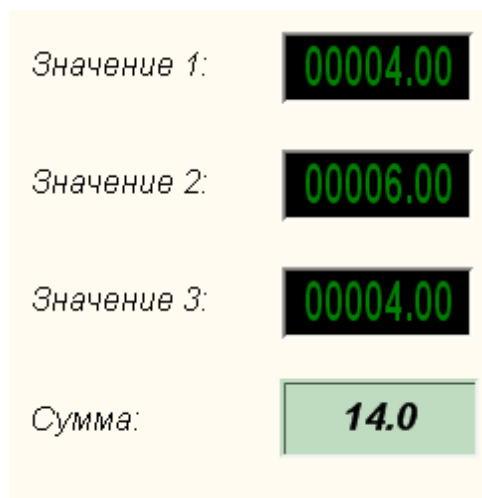
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Сумматор](#)<sup>[160]</sup> складывает все входные значения, которое задается [Селекторами](#)<sup>[213]</sup>. [Таймер](#)<sup>[183]</sup> подает синхроимпульсы на вход [D-триггеров](#)<sup>[384]</sup>, чтобы обеспечить периодичность поступления численных значений в [Сумматор](#)<sup>[160]</sup> (т.к. сложение сигналов происходит только при одновременном изменении значений сигналов на входных контактах сумматора). [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления информации.

### *Результат работы проекта*




Данный пример непосредственно в ZETView

## 4.18. Умножение

**Умножение** - компонент. Служит для реализации такой арифметической операции, как нахождения произведения двух чисел.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Вход 1 - канал №1 является первым множителем.
- Вход 2 - канал №2 является вторым множителем.

## ➤ Выходные

- Выход - выходной канал, на котором формируются произведение численных значений каналов.

**Настраиваемые свойства:**

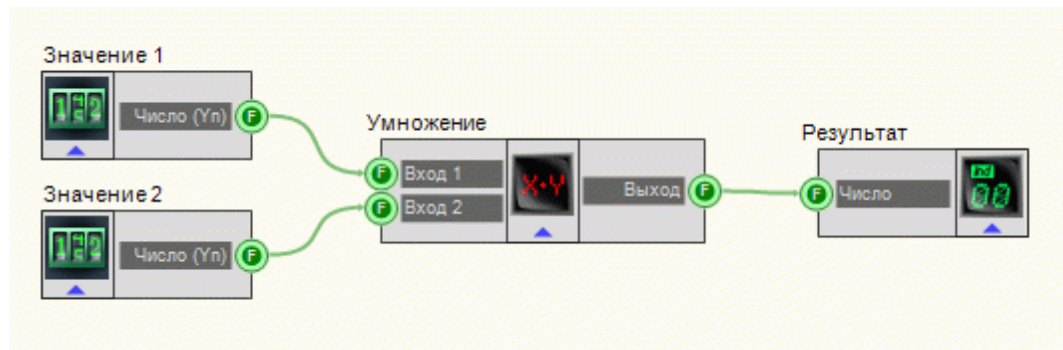
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

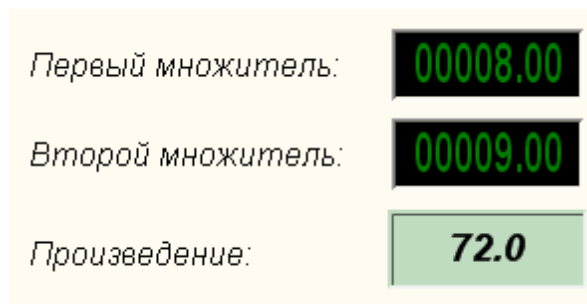
- FirstReaction (false) - установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (false) - установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- First (0) - значение по умолчанию первого операнда.
- Second (0) - значение по умолчанию второго операнда.

**Пример****Проект в SCADA ZETView**



В этой схеме компонент [Умножение](#)<sup>[161]</sup> производит операцию нахождения произведения двух чисел. [Селекторы](#)<sup>[213]</sup> служат для ввода численных значений на вход компонента. [Цифровой индикатор](#)<sup>[362]</sup> нужен для графического вывода результирующего численного значения.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 4.19.Усреднение

**Усреднение** - компонент. Служит для реализации такой математической операции, как нахождения среднеарифметического или среднеквадратического всех значений, приходящих на вход, за заданный промежуток времени.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

➤ Входящие:

- Значение - контакт, на который поступают числа, для которых нужно найти их среднее арифметическое или среднее геометрическое значения.
- Время - контакт, на который подается значение времени в секундах [с], за которое будет производиться усреднение.

➤ Выходные:

- Среднее - контакт, на котором формируются результирующее усредненное значение.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- AverageTime (1) - установка времени усреднения, с.
- averagetype (CA3) - тип усредненного значения:
  1. CA3 - среднее арифметическое значение.
  2. СКЗ - среднее квадратическое значение.

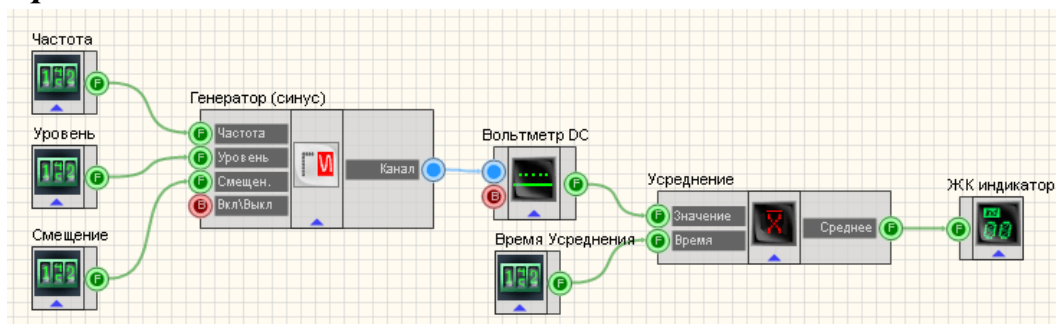


## Пример



*Данный пример требует подключенного ZET-устройства, содержащего ЦАП.*

### Проект в SCADA ZETView



В проекте реализована схема работы компонента "[Усреднение](#)<sup>[163]</sup>". Компонент "[Генератор \(синус\)](#)<sup>[247]</sup>" принимая сигнал с физического ЦАП ZET-устройства, подключенного к ПК, формирует сигнал по заданным параметрам. Параметры задаются с помощью соответствующих [селекторов](#)<sup>[213]</sup>, подключенных к [Генератору \(синус\)](#)<sup>[247]</sup>. Далее [Вольтметр постоянного тока](#)<sup>[288]</sup> снимает мгновенные значения напряжения и отправляет их на компонент "[Усреднение](#)<sup>[163]</sup>". С помощью селектора, названного "Время

усреднения", выставляется время, в течение которого компонент будет принимать значения с вольтметра для усреднения. Результат выводится на [ЖК индикатор](#)<sup>[348]</sup>.

### *Результат работы проекта*



При запуске проекта усредненное значение примет неменяющуюся величину, если изменить значение смещения сигнала на [Генераторе\\_\(синус\)](#)<sup>[247]</sup>, значение усреднения изменится. По умолчанию в данном примере компонент рассчитывает среднее арифметическое.

Посмотреть пример непосредственно в ZETView

## Глава 5. Внешние устройства

### 5.1. Акустический модем

<Раздел находится в разработке>

**Акустический модем** - компонент. Предназначен для преобразования цифровых сигналов в звуковые сигналы речевого диапазона и обратно. Устройство способное передавать и принимать заданные сигналы звуковых частот по телефонным линиям, что позволяет компьютеру и терминалу соединиться используя модем и обычный телефон.

#### **Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------

	Не имеет
---	----------

### Параметры:

#### ➤ Входные:

- Время - установка времени АК.
- Всплытие - установка времени всплытия.
- Синхро - получить данные.
- Уст. время - установить время.
- Сон - переключить модем в режим сна.
- Перезагр.- перезагрузить модем.

#### ➤ Выходные

- Время - текущее время АК.
- Всплытие - текщее время всплытия.
- Статус - состояние связи с модемом.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Device (0) - устройство

## 5.2.Весы ВР05мс

<Раздел находится в разработке>

**Весы ВР05мс-** компонент. Предназначен для взвешивания.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет



**Параметры:**

## ➤ Входящие:

Не имеет.

## ➤ Выходные

- Вес - выходные значения весов.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

Не имеет.

### 5.3. Генератор DS360

<Раздел находится в разработке>

**Генератор DS360** - компонент. Используются для воспроизведения электромагнитных сигналов. Прибор DS360 позволяет воспроизводить сигнал определенной формы, что бывает необходимо при исследованиях и настройке различных электронных систем. Устройство позволяет формировать определенные виды импульсов с заданными характеристиками. С помощью него можно легко и удобно провести испытания вычислительных устройств.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Частота - частота генерируемого сигнала, Гц.
- Уровень - уровень генерируемого сигнала, В
- Смещение - смещение генерируемого сигнала, В
- Вкл/выкл - состояние работы генератора.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- PortNumber (1) - номер последовательного (COM) порта для работы с генератором
- BaudRate (19200) - скорость обмена данными по последовательному (COM) порту с генератором, бит/с
- signaltype (Синус) - тип генерируемого сигнала:
  1. Синус.
  2. Меандр.
- Frequency (1000) - частота генерируемого сигнала, Гц.
- Amplitude (1) - уровень СКЗ генерируемого сигнала, В
- Offset (0) - смещение нуля генерируемого сигнала, В
- OutputStatus (false) - состояние работы генератора.

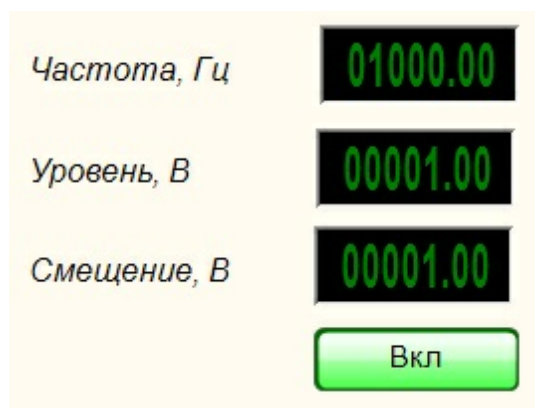


### Пример

#### Проект в SCADA ZETView



#### Результат работы проекта



Данный пример непосредственно в ZETView

## 5.4.Источник питания LPS-305

<Раздел находится в разработке>

**Источник питания LPS-305** - компонент. Предназначен для управления программируемым источником питания LPS-305.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- U1 - ограничение по напряжению первого канала, В.
- I1 - ограничение по току первого канала, А.
- U2 - ограничение по напряжению второго канала, В.
- I2 - ограничение по току второго канала, А.
- Вкл/Выкл - включение/выключение источника питания.

➤ Выходные

- Напр.1 - текущее значение по напряжению первого канала, В.
- Ток 1 - текущее значение по току первого канала, А.
- Напр.2 - текущее значение по напряжению второго канала, В.
- Ток 2 - текущее значение по току второго канала, А.

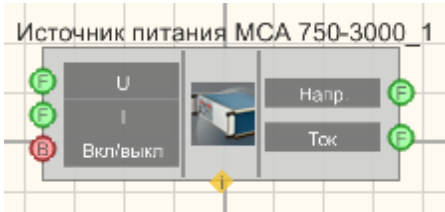
#### Настраиваемые свойства:

- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>92</sup>
- Частные свойства:
  - Не имеет.

## 5.5.Источник питания MCA 750-3000

**Источник питания MCA 750-3000** - компонент. Предназначен для управления программируемым источником питания **MCA 750-3000**.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

- Входные:
  - U - ограничение по напряжению, В.
  - I - ограничение по току канала, А.
  - Вкл/Выкл - включение/выключение источника питания.
- Выходные
  - U - текущее значение по напряжению, В.
  - I - текущее значение по току канала, А.

#### Настраиваемые свойства:

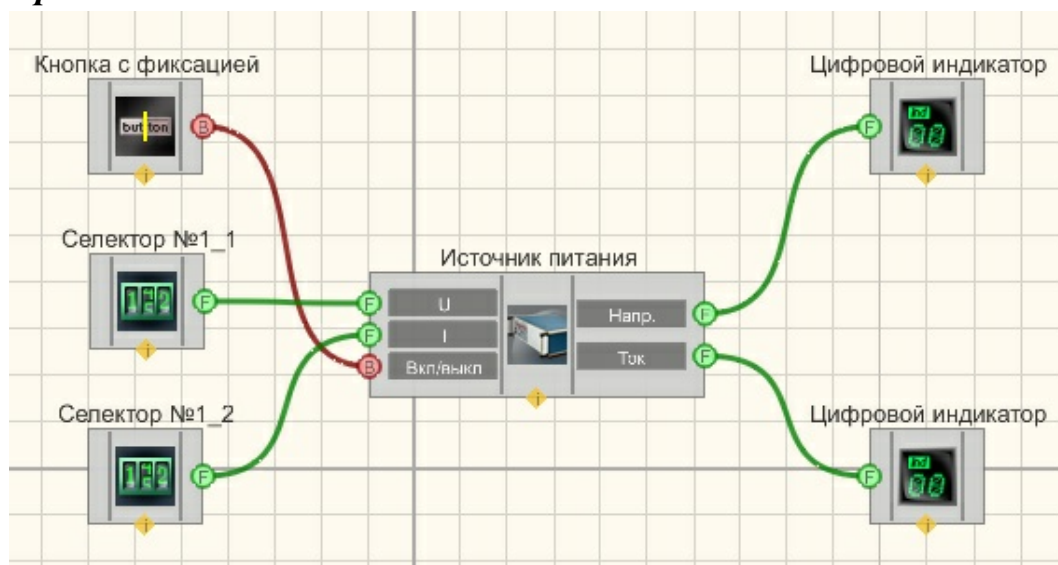
- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>92</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):

- Power (false):
  1. False - выключение;
  2. True - включение.
- Volt (0) - установка значения напряжения;
- Curr (0) - установка значения тока;
- PortNumber (1) - номер последовательного (COM) порта для работы с источником питания

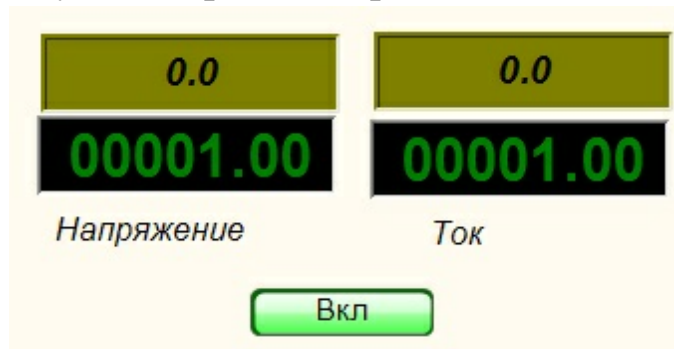


## Пример

### Проект в SCADA ZETView



### Результат работы проекта



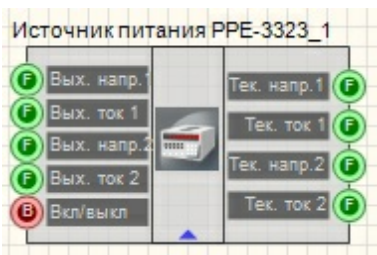
Данный пример непосредственно в ZETView

## 5.6.Источник питания PPE-3323

<Раздел находится в разработке>

**Источник питания PPE-3323** - компонент. Предназначен для управления программируемым источником питания PPE-3323.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вых. напр. 1- значение выходного напряжения источника питания по первому каналу, В.
- Вых. ток 1 - значение выходного тока источника питания по первому каналу, А.
- Вых. напр. 2- значение выходного напряжения источника питания по второму каналу, В.
- Вых. ток 2 - значение выходного тока источника питания по второму каналу, А.
- Вкл/Выкл - включение/выключение источника питания.

#### ➤ Выходные

- Тек. напр.1 - текущее значение выходного напряжения источника питания по первому каналу, В.
- Тек. ток 1 - текущее значение выходного тока источника питания по первому каналу, А.
- Тек. напр.2 - текущее значение выходного напряжения источника питания по второму каналу, В.
- Тек. ток 2 - текущее значение выходного тока источника питания по второму каналу, А..

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

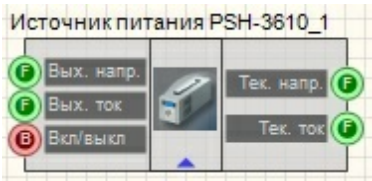
- PortNumber (1) - номер последовательного (COM) порта для работы с источником питания
- Voltage1 (0) - значение выходного напряжения источника питания по первому каналу, В.
- CurrentU1 (0) - значение выходного тока источника питания по первому каналу, А
- Voltage2 (0) - значение выходного напряжения источника питания по второму каналу, В.
- CurrentU2 (0) - значение выходного тока источника питания по второму каналу, А
- OVPValue1 (34) - значение защитного уровня напряжения по первому каналу, В
- OVPValue2 (34) - значение защитного уровня напряжения по второму каналу, В
- OCPStatus (false) - защита выходов от перегрузки.
- digitalvoltage (3.3)- значение выходного напряжения источника питания по третьему каналу.
- OutputStatus (false) - состояние работы источника питания.

## 5.7. Источник питания PSH-3610

<Раздел находится в разработке>

**Источник питания PSH-3610** - компонент. Предназначен для управления программируемым источником питания PSH-3610.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Вых. напр. - значение выходного напряжения источника питания, В
- Вых. ток - значение выходного тока источника питания, А
- Вкл/выкл - состояние работы источника питания.

➤ Выходные

- Тек. напр. - текущее значение выходного напряжения источника питания, В

- Тек. ток - текущее значение выходного тока источника питания, А

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

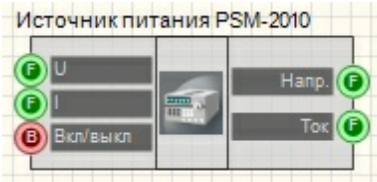
- PortNumber (1) - номер последовательного (COM) порта для работы с источником питания
- BaudRate (9600) - скорость обмена данными по последовательному (COM) порту с источником питания, бит/с
- Voltage (0) - значение выходного напряжения источника питания по каналу, В.
- Current (0) - значение выходного тока источника питания по каналу, А
- OVPValue (38) - значение защитного уровня напряжения по каналу, В
- OCPStatus (false) - защита выходов от перегрузки.
- OutputStatus (false) - состояние работы источника.

## 5.8.Источник питания PSM-2010

<Раздел находится в разработке>

**Источник питания PSM-2010** - компонент. Предназначен для управления программируемым источником питания PSH-2010.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- U - ограничение по напряжению канала, В
- I - ограничение по току канала, А
- Вкл/выкл - состояние работы источника питания.

##### ➤ Выходные

- Напр. - текущее значение напряжения по каналу, В



- Ток - текущее значение тока по каналу, А

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства:

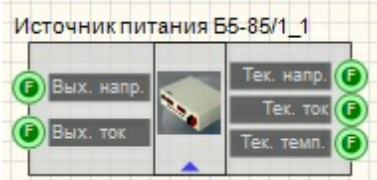
Не имеет.

## 5.9.Источник питания Б5-85/1

<Раздел находится в разработке>

**Источник питания Б5-85/1** - компонент. Предназначен для управления программируемым источником питания Б5\_85/1.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входные:

- Вых. напр - значение выходного напряжения источника питания, В
- Вых. ток - значение выходного тока источника питания, А

➤ Выходные

- Тек. напр. - текущее значение выходного напряжения источника питания, В
- Тек. ток - текущее значение выходного тока источника питания, А
- Тек. темп. - текущее значение температуры внутри источника питания.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

PortNumber (1) - номер последовательного (COM) порта для работы с источником питания

- Voltage (0) - значение выходного напряжения источника питания по первому

каналу, В.

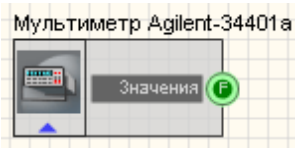
- Current (0) - значение выходного тока источника питания по первому каналу, А

## 5.10.Мультиметр Agilent 34401a

<Раздел находится в разработке>

**Мультиметр Agilent 34401a** - компонент. Предназначен для измерения параметров постоянного и переменного тока, частоты переменного тока, сопротивления участка цепи, для прозвона цепи, для проверки р-п переходов.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Выходные

- Значения - выходные значения мультиметра.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

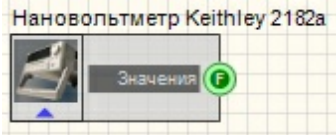
- PortNumber (1) - номер последовательного (COM) порта для работы с генератором
- BaudRate (9600) - скорость обмена данными по последовательному (COM) порту с генератором, бит/с
- measuretype (постоянное напряжение) - тип измеряемого значения:
  1. постоянное напряжение.
  2. отношение постоянных напряжений.
  3. переменное напряжение.
  4. постоянный ток.
  5. переменный ток.
  6. сопротивление (2 пр.).
  7. сопротивление (4 пр.)
  8. частота.
  9. период.

10. непрерывность электрических цепей.
11. испытание диодов.

## 5.11. Нановольтметр Keithley 2182a

**Нановольтметр Keithley 2182a** - компонент. Предназначен для измерения сверхнизких напряжений, имеет низкий уровень собственных шумов и обеспечивает высокую скорость измерений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Выходные

- Значения - выходные значения нановольтметра.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- PortNumber (1) - номер последовательного (COM) порта для работы с генератором
- BaudRate (9600) - скорость обмена данными по последовательному (COM) порту с генератором, бит/с
- Channel (DCV2) - измеряемый канал.
- range (10) - диапазон измерений, мВ.
- rate (Fast) - скорость обновления данных в устройстве, возможно 3 варианта скорости: Fast (быстрый), Medium (средний), Slow (медленный)

## Глава 6.Время

### 6.1.Временная задержка

**Временная задержка** - компонент. Предназначен для формирования задержки передачи входных данных на заданную величину.

Зачастую данный компонент используется как вспомогательный элемент в схемах, где на один компонент поступает несколько потоков данных. Так как данные проходят цепочки компонентов различной длины и проходят разные стадии обработки, требующие затрат некоторого времени, на входе принимающего компонента может возникнуть рассинхронизация. Потоки данных, которые должны приходить одновременно могут запаздывать. Это может вызвать конфликты в работе проекта. Наглядный пример можно найти в описании компонента "[Гистограмма](#)"<sup>[452]</sup>.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Вход - входной канал, в котором требуется организовать задержку.

##### ➤ Выходные:

- Выход - выход, на котором появится сигнал со входа только по истечению времени задержки.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

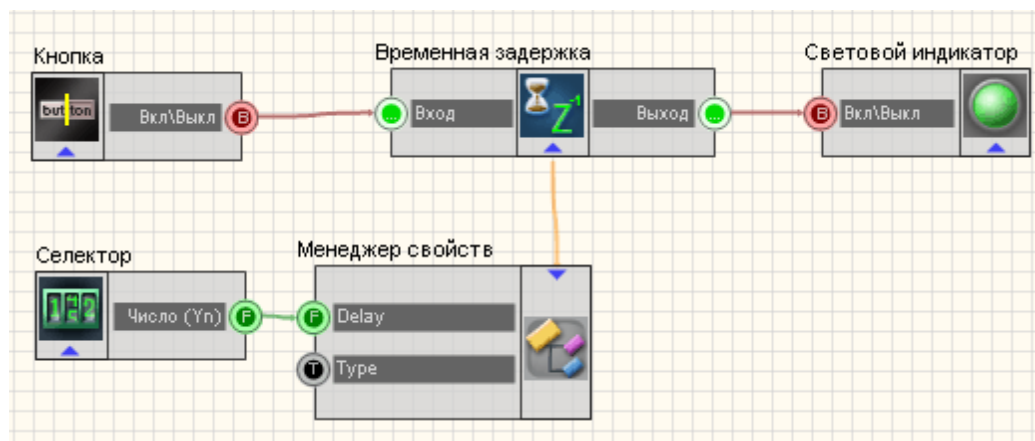
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Delay (1000) - значение времени задержки, мс.
- type (по времени) - установка типа задержки:
  1. По времени - передача значения, пришедшего на вход, по истечении указанного времени.
  2. На один такт - значение, пришедшее на вход, будет передано с выхода в момент времени, когда на вход поступит следующее значение.



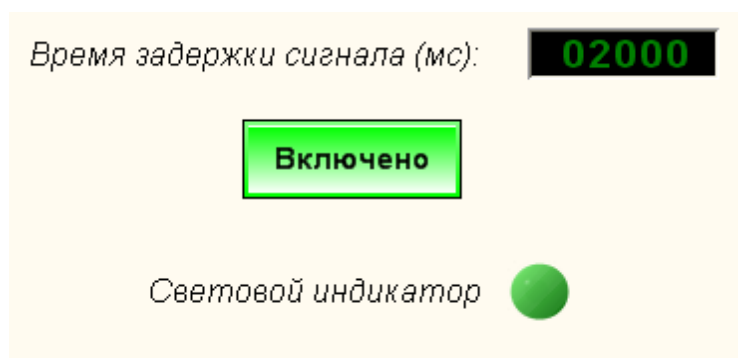
## Пример

### Проект в SCADA ZETView



В представленной схеме компонент "[Временная задержка](#)"<sup>[178]</sup> формирует запаздывание сигнала, поступающего с [кнопки с фиксацией](#)<sup>[367]</sup>. Через [менеджер свойств](#)<sup>[534]</sup> можем выставить свойство "Delay"(время задержки) компонента "[Временная задержка](#)"<sup>[178]</sup> с помощью подключенного к нему [селектора](#)<sup>[213]</sup>. В результате [световой индикатор](#)<sup>[356]</sup> гаснет и загорается с запаздыванием.

### Результат работы проекта



Выставить время задержки 2000 мс, нажать кнопку. Световой индикатор загорится с запаздыванием 2 секунды.

Данный пример непосредственно в ZETView

## 6.2.Время сервера

**Время сервера** - компонент. Позволяет получать время канала.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Канал - канал, время которого требуется определить.
- Синхро - контакт, при подаче на который высокого логического уровня происходит опрос времени канала.

➤ Выходные

- Время - вывод числа секунд, прошедшее со времени включения канала.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

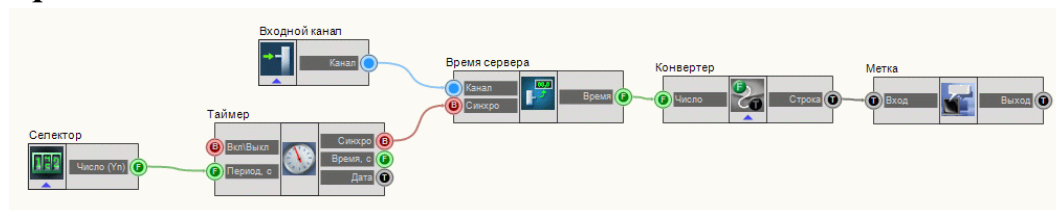
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Auto (false) - включение и выключение автоматического опроса времени.
- Frequency (10) - частота опроса, Гц.



### Пример

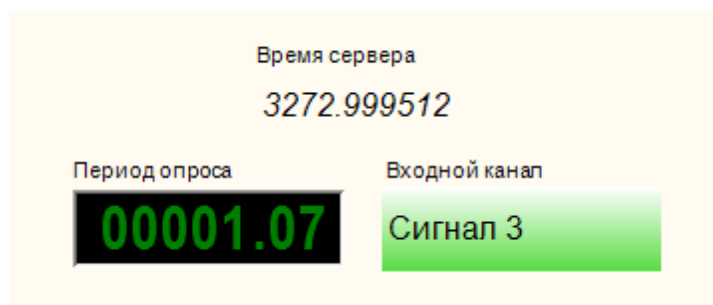
#### Проект в SCADA ZETView



В этой схеме компонент [Время сервера](#)<sup>[180]</sup> позволяет определить время канала. [Таймер](#)<sup>[183]</sup> нужен для формирования синхроимпульсов, по

которым будет происходить опрос канала. [Селектор](#)<sup>[213]</sup> служит для задачи периодичности создания синхроимпульсов. [Конвертер](#)<sup>[379]</sup> необходим для преобразования численного типа в строковый. [Метка](#)<sup>[554]</sup> служит для графического представления информации.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## 6.3. Информация о дате

**Информация о дате** - компонент. Позволяет получать текущую дату при подаче на вход компонента управляющего воздействия.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
<p>The screenshot shows the 'Информация о дате' component in design mode. It features a central display area with a red 'В' (Input) button and a 'Импульс' (Pulse) label. To the right, there is a vertical list of date-related parameters, each with a corresponding icon: Число (F), Месяц (F), Месяц (T), Год (F), Тек. дата (T), Сп. дата (T), and Время (T).</p>	Не имеет

### Параметры:

#### ➤ Входы:

- Импульс - вывод, при подаче на который управляющего воздействия, с выходных контактов поступает информация о текущей дате.

#### ➤ Выходные:

- Число - вывод текущего календарного дня в численном виде.

- Месяц - вывод текущего месяца в численном виде.
- Месяц - вывод текущего месяца в виде текстовой строки.
- Год - вывод текущего года в численном виде.
- Текущая дата - вывод текущей даты в виде текстовой строки (ДД.ММ.ГГ).
- Следующая дата - текущая дата с постоянно прибавляемым к ней числом добавочных месяцев и лет (ДД.ММ.ГГ).
- Время - вывод текущего времени.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

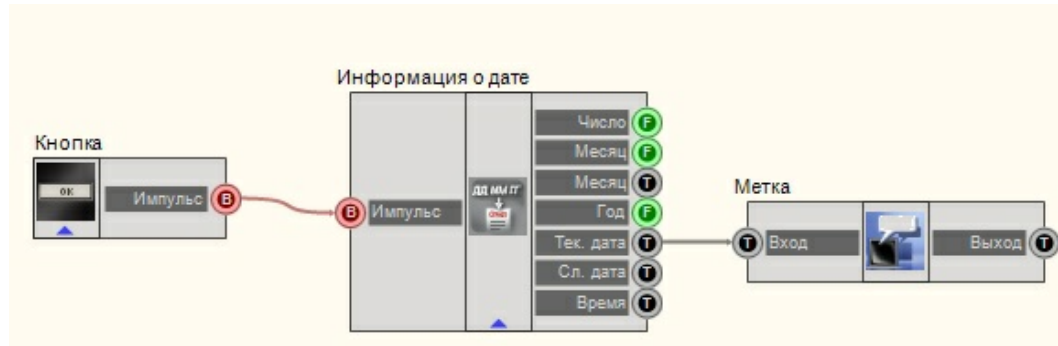
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Declination (false) - включение/отключение склонения месяцев.
- AddMonth (0) - число добавочных месяцев к текущей дате.
- AddYear (0) - число добавочных лет к текущей дате.



### Пример

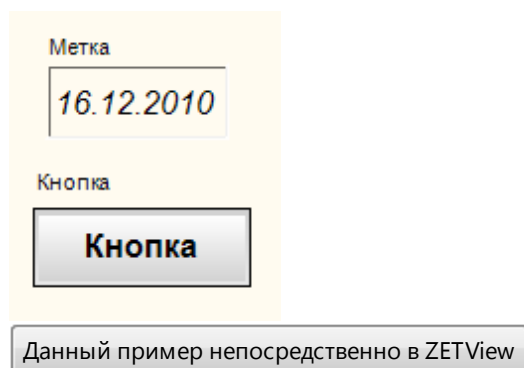
#### Проект в SCADA ZETView



В этой схеме компонент [Информация о дате](#)<sup>[181]</sup> передает на входной канал [Метки](#)<sup>[554]</sup> текущую дату. [Кнопка](#)<sup>[367]</sup> служит для посылки на вход [Информации о дате](#)<sup>[181]</sup> управляющего импульса, для того, чтобы осуществилась передача даты в выходные каналы компонента.

### Результат работы проекта





## 6.4. Таймер



### *Пример*

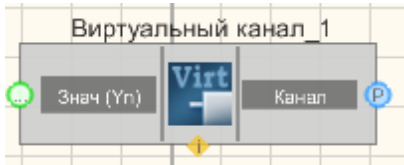
*Пример* создания проекта, в котором производится инкрементирование численных значений канала: (находится в Примере компонента [Инкрементирование](#)<sup>[140]</sup>)

## Глава 7. Вход (АЦП)

### 7.1. Виртуальный канал

**Виртуальный канал** - компонент. Специальный канал, эмулирующийся внутри ZETView. Является аналогом модуля [Входного сигнала](#)<sup>[185]</sup>, с той разницей, что модуль Входного сигнала получает информацию с реального физического устройства, а Виртуальный канал выводит смоделированный самим компьютером сигнал.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные

- Знач (Yn) - входное значение записываемое в созданный виртуальный канал.

#### ➤ Выходные

- Канал - выходной сигнал.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Name (Канал) - цифробуквенное обозначение канала.
- Conversion (mB) - единица измерения по каналу.
- Frequency (50000) - Частота дискретизации по каналу, Гц.
- MaxLevel (1000) - максимально допустимый уровень (в единицах измерения канала).
- MinLevel (0) - минимально допустимый уровень (в единицах измерения канала).
- Reference (0.001) - опора для расчета уровня в дБ (в единицах измерения канала, нулевой уровень для начала отсчета шкалы в дБ).

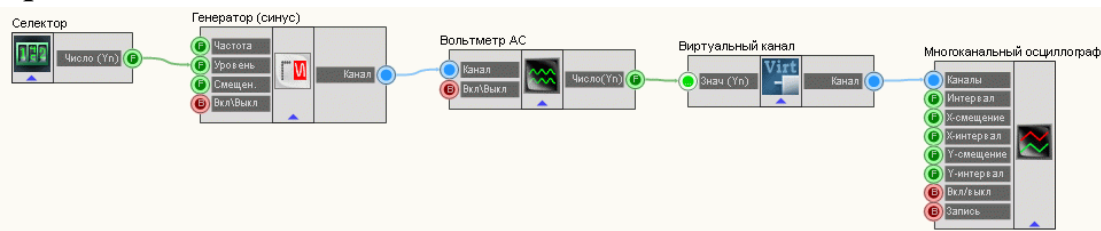


### Пример



*Данный пример требует подключенного ZET-устройства, содержащего ЦАП.*

### Проект в SCADA ZETView



В этой схеме компонент [Виртуальный канал](#)<sup>[183]</sup> преобразует набор среднеквадратичных значений, снятых с [Вольтметра переменного тока](#)<sup>[284]</sup> в канал, значения которого при помощи [Многоканального осциллографа](#)<sup>[494]</sup> выводятся на экран. [Селектор](#)<sup>[213]</sup> нужен для установки амплитуды синусоидального сигнала, получаемого при помощи [Синусоидального генератора](#)<sup>[247]</sup>.

### *Результат работы проекта*



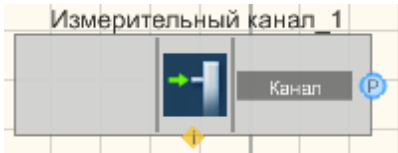

Данный пример непосредственно в ZETView

## 7.2. Измерительный канал

**Измерительный канал** - компонент. Позволяет принимать сигнал с выхода периферийного устройства. Имеет единственный параметр - выбранный канал, с которого поступает сигнал. Без данного модуля невозможно принимать данные с внешних устройств.



*Использование компонента требует подключенного ZET-устройства, содержащего ЦАП.*

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Канал - предоставляет данные, с которыми будет оперировать компонент, подключенный к нему (данные с канала АЦП, виртуальный ЦАП).

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Channel (вх.канал) - установка имени канала.
- ActiveColor (a0e1a0)- настройка цвета фона в активном состоянии. По умолчанию зеленый.
- InactiveColor (e1a0a0)- настройка цвета фона в неактивном состоянии. По умолчанию красный.
- showlist (Да)- включение/отключение показа выпадающего списка с каналами.
- NewStyle (false) - новый стиль отображения.

При двойном нажатии левой кнопкой "мыши" на компоненте Измерительный канал в Интерфейсе оператора на этапе проектирования проекта и при нажатии левой или правой кнопкой "мыши" на компоненте во время выполнения проекта появляется меню выбора канала:



При добавлении компонента в проект он не передает с выхода никакого значения. При выборе измерительного или виртуального канала, компонент отображается зеленым цветом, если канал существует в списке каналов сервера. Если установленным является

несуществующий канал, компонент изменяет свой цвет на красный:

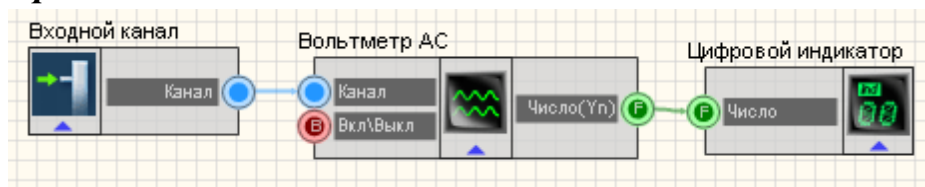
## Соединительные ножки

Контакт №1 - "Канал". Предназначен для передачи ID (идентификационный номер) канала сервера подключенным компонентам. Данный контакт поддерживает множественное соединение и является выходным. Интерфейс контакта – «Целое число».



### Пример

#### Проект в SCADA ZETView



В этой схеме компонент передает данные с выбранных каналов. Далее чтобы графически представить информацию на дисплей, с [Вольтметр переменного тока](#)<sup>[284]</sup> преобразованный сигнал идет на [Цифровой индикатор](#)<sup>[362]</sup>. Результат работы составленной программы можно увидеть в интерфейсе оператора.

#### Результат работы проекта

Выбор входного канала:

Сигнал 1

Значение напряжения  
на входном канале:

0.5

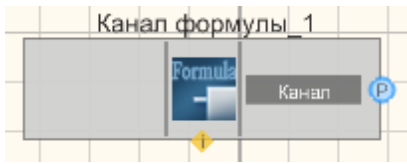
Данный пример непосредственно в ZETView

## 7.3. Канал формулы

<Раздел находится в разработке>

**Канал формулы** - компонент. Позволяет не только формировать сигналы различной формы, но и производить над ними различные математические и арифметические операции.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

Не имеет.

## ➤ Выходные

- Канал - виртуальный выходной канал, с которого поступают мгновенные значения, получаемые с формулы.

**Настраиваемые свойства:**

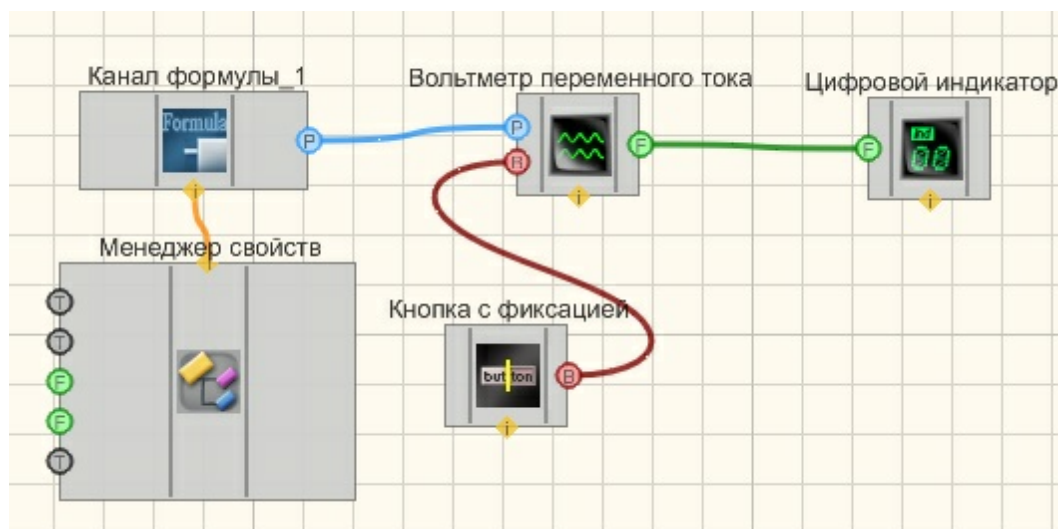
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

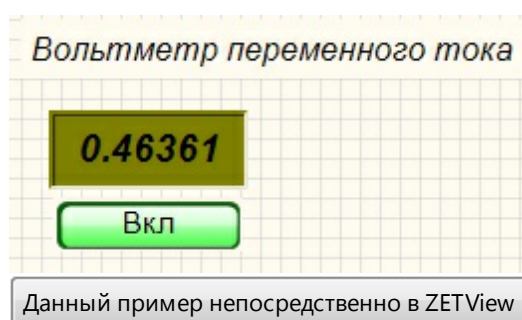
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ChannelName (канал) - цифробуквенное обозначение канала.
- Conversion (мВ) - единица измерения по каналу.
- MaxLevel (1000) - максимально допустимый уровень по каналу.
- Reference (0.001) - опора для расчета уровня в дБ (нулевой уровень для начала отсчета шкалы в дБ).
- Formula (0) - формула

***Пример******Проект в SCADA ZETView***



### *Результат работы проекта*



## 7.4.Опрос параметров канала

**Опрос параметров канала** - компонент. Позволяет в определенный момент по внешнему воздействию или с определенной частотой снимает параметры сигнала с выбранного канала.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - подключение канала, параметры которого следует опросить.
- Импульс - канал, на котором при изменении логического состояния (с низкого логического уровня на высокий) произойдет опрос входного канала.

#### ➤ Выходные

- Т общее - вывод общего (системного времени) канала.
- Т канала - вывод времени сигнального процессора канала.
- Частота - выход компонента, который выдает частоту дискретизации по входному каналу.
- Единица измерения - выдает единицы измерения канала, которые настроены в меню Редактирование файлов параметров.
- Максимальный уровень - выдает максимально допустимый уровень по каналу.
- Минимальный уровень - выдает минимально допустимый уровень по каналу.
- Опора - опорное значение, относительно которого считается уровень сигнала в дБ.
- Смещение - выдает смещение постоянной составляющей
- Файл - вывод имени файла, в котором хранятся дополнительные или поправочные данные о канале.
- Чувствительность - позволяет определить чувствительность преобразований канала (В/единицу измерения).
- Статус - позволяет определить работает ли канал.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):



- WorkingTime (0) - глобальное время ZETServera, с.
- ChannelTime (0) - текущее время по каналу, с.
- synchrotype (по запросу) - способы опроса канала (два способа).
  1. По внешнему запросу.
  2. Автоопрос 1 раз в секунду.
- Timeout (5) - время, по истечению которого выдается статус канала, когда время по нему не течет, с.

**Примечание.** В компоненте <Опрос параметров сигнала> есть ножка "Т общее", с которой идет абсолютное время канала, т.е. время с момента последней перестартовки ZETServer.

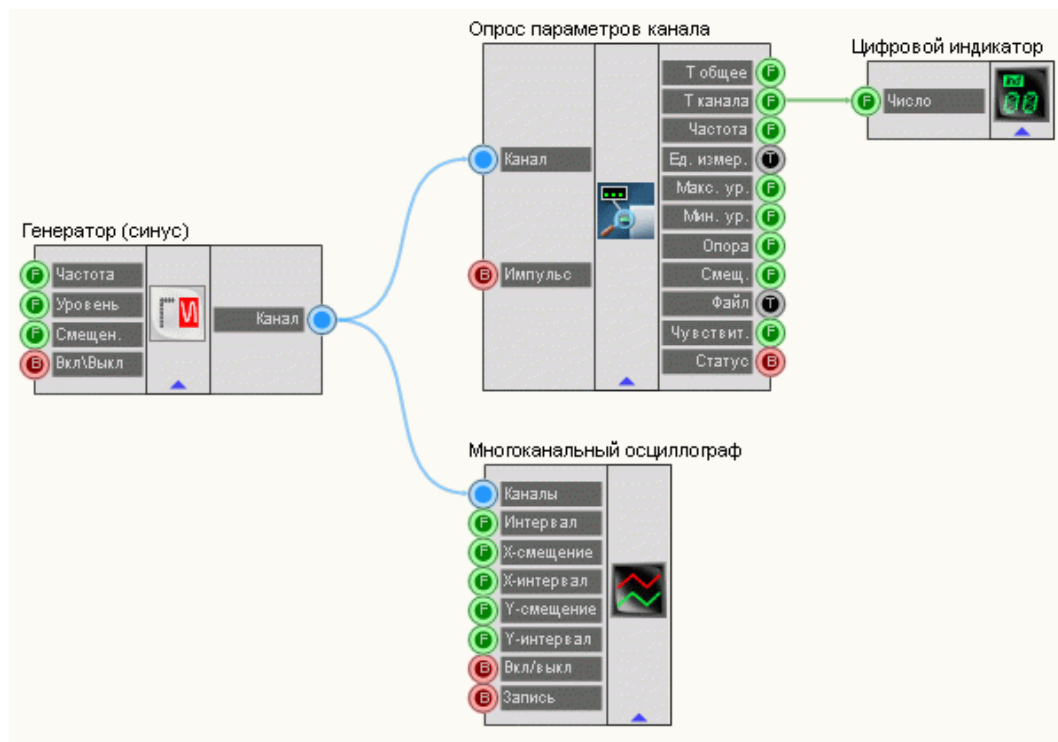


### Пример



*Данный пример требует подключенного ZET-устройства, содержащего ЦАП.*

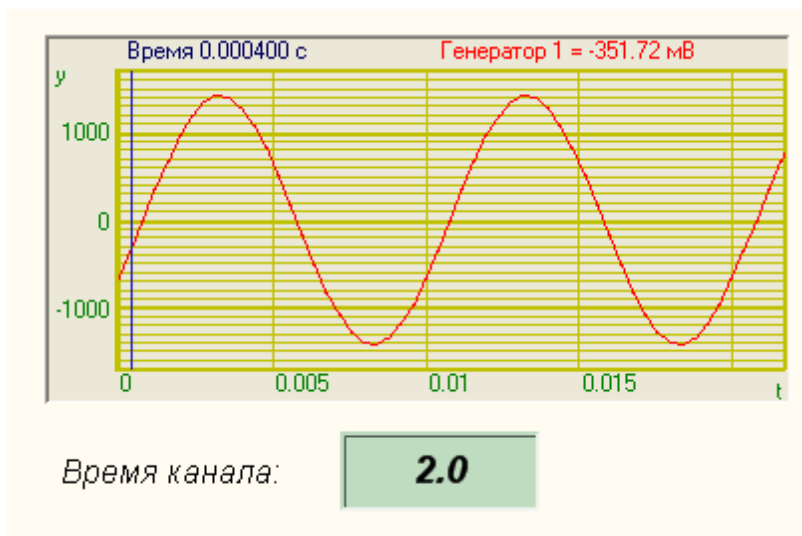
### Проект в SCADA ZETView



В этой схеме с [Синусоидального сигнала](#)<sup>247</sup> гармонический сигнал

поступает на [Опрос параметров канала](#)<sup>[189]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> служит для отображения времени канала на дисплее. [Многоканальный осциллограф](#)<sup>[494]</sup> необходим для просмотра сигнала, поступающего с генератора.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

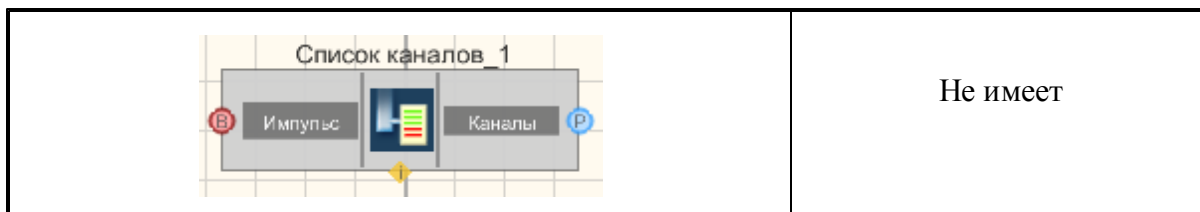
## 7.5.Список каналов

**Список каналов** - компонент. Компонент "Сортировка каналов" предназначен для сортировки каналов. **Список каналов** для сортировки может быть задан в окне настройки свойств или выбран оператором в процессе работы проекта. Вид и условия сортировки указываются при настройке компонента.

**Список каналов** - осуществляет выбор списка каналов оператором и передача ID выбранных каналов подключенным компонентам. При получении компонентом импульса, открывается диалоговое окно, в котором оператор выбирает каналы для отображения. При закрытии окна кнопкой ОК, список выбранных каналов передаётся подключенным компонентам. Дополнительно компонент **Список каналов** реализует функции фильтра выбранных каналов по названию и/или по единицам измерения.

#### **Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Импульс - импульс для корректировки списков каналов.

#### ➤ Выходные

- Каналы - на выходе отсортированный список каналов.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Channels - список каналов для сортировки (перечисляются через разделитель "точка с запятой").
- SortName (false) - разрешение сортировки каналов по части имени.
- Name (Сигнал) - часть имени канала, по которой происходит сортировка каналов при разрешении сортировки по части имени канала.
- SortConversion (false) - разрешение сортировки каналов по единице измерения по каналу.
- Conversion (мВ) - единица измерения по каналу.

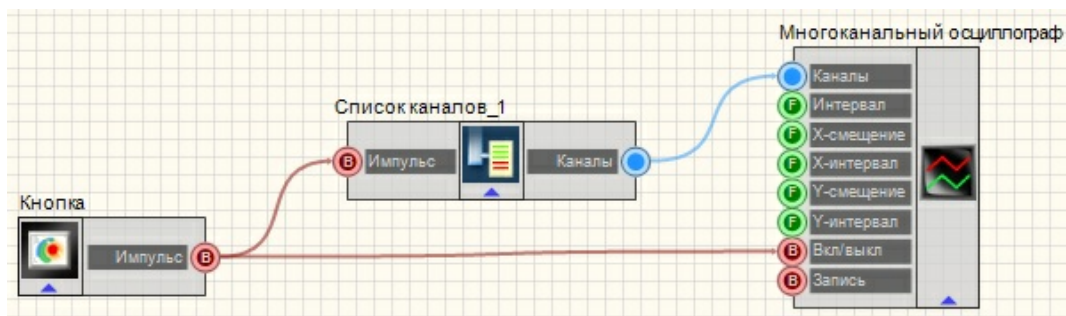
## Работа с компонентом

После запуска проекта при получении компонентом "Сортировка каналов" импульса (например, от подключенной кнопки) открывается окно "Список каналов", в котором отображаются каналы, выбранные для сортировки. Если не выбран ни один канал, поле остаётся пустым. При нажатии правой кнопкой "мыши" в поле "Список каналов" появляется контекстное меню, в котором отображаются все каналы zet-сервера. При выборе какого-либо канала, он добавляется в список для сортировки. При выборе пункта Все каналы в список добавляются все каналы zet-сервера. При закрытии окна кнопкой "ОК" список выбранных каналов сортируется и компонент передает с выхода список каналов, удовлетворяющих условиям сортировки. При закрытии окна кнопкой "Отмена" изменение списка игнорируется и сортировка не производится.

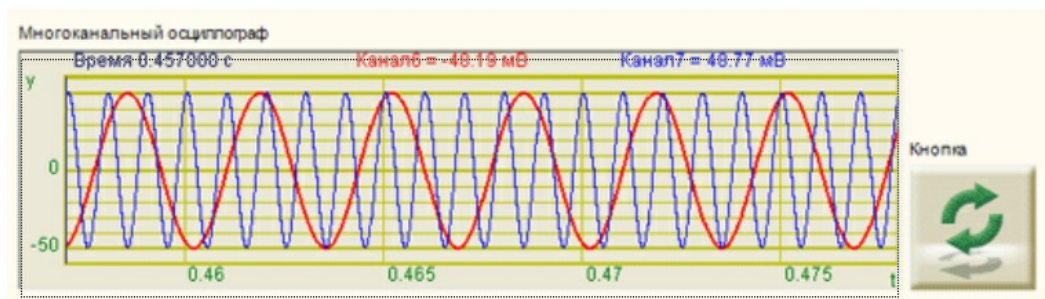


### Пример

## Проект в SCADA ZETView



На рисунках ниже приведен пример реализации многоканального осциллографа. Используемые компоненты: Кнопка с картинкой, Сортировка каналов, Многоканальный осциллограф. Компонент Сортировка каналов используется в данном случае для создания списка каналов для отображения, поскольку простого подключения нескольких каналов к Многоканальному осциллографу недостаточно для их одновременного отображения.



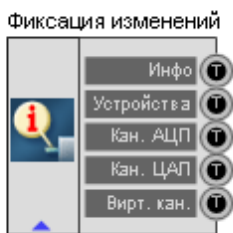
### *Результат работы проекта*

Данный пример непосредственно в ZETView

## 7.6. Фиксация изменений

**Фиксация изменений** - компонент. Позволяет фиксировать изменения, происходящие с устройствами, каналами внутри системы, обращаясь для этого к компоненту ZETServer.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Выходные:

- Информация - выводит, что произошло.
- Устройства - выводит имя устройства, с которым происходят изменения.
- Канал АЦП - выводит имя канала АЦП, с которым происходят изменения.
- Канал ЦАП - выводит имя канала ЦАП, с которым происходят изменения.
- Виртуальный канал - выводит имя виртуального канала, с которым происходят изменения.

**Настраиваемые свойства:**

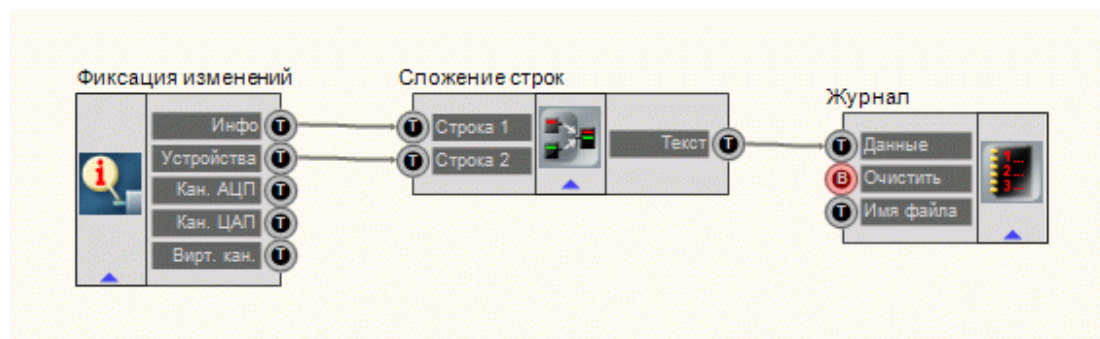
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ADCChannels (true) - мониторинг каналов АЦП.
- DACChannels (true) - мониторинг каналов ЦАП.
- VirtualChannels (true) - мониторинг виртуальных каналов.
- Devices (true) - мониторинг внешних устройств.

**Пример****Проект в SCADA ZETView**



В этой схеме компонент [Фиксация изменений](#)<sup>[194]</sup> регистрирует изменения происходящие с одним из внешних подключаемых модулей в компонент [Журнал событий](#)<sup>[546]</sup>. Но после предварительной склеивания информации об устройстве и происходящими с ним изменениями (компонент [Сложение строк](#)<sup>[558]</sup>).

### *Результат работы проекта*

Событие
A19U2 №803;
BC-110 - подключенаA19U2 №803;
BC-401 - подключенаA19U2 №803;
Устройство A17U2 №803 подключеноA19U2 №803;
Устройство A17U2 №803 подключеноA19U2 №803;
Устройство A17U2 №803 подключеноA19U2 №803;
BC-112 - подключен
BC-110 - подключен
BC-120 - подключен
BC-130 - подключен
Устройство A17U8 №289 подключено
Устройство A17U8 №289 подключеноZET017U4 №289;
Устройство A17U8 №289 подключеноZET017U4 №289;
Устройство A17U8 №289 подключеноZET017U4 №289;

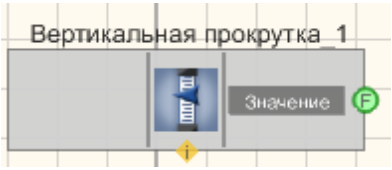
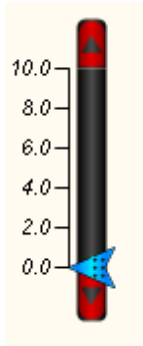
Данный пример непосредственно в ZETView

## Глава 8.Выбор

### 8.1.Вертикальная прокрутка

**Вертикальная прокрутка** - компонент. Предназначен для выбора значения посредством перемещения указателя на шкале и последующей передачи значения в канал.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

#### Параметры:

##### ➤ Выходные:

- Значение - вывод, с которого идет управляющий сигнал.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства(в скобках значение, установленное по умолчанию):

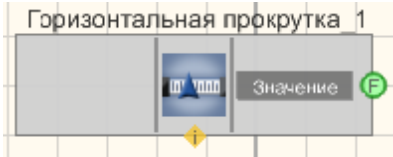
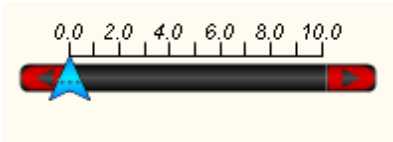
- TextSize (8) - размер текста.
- textstyle (FontStyleItalic) - стиль текста
- CurrentValue (0) - установка текущего значения.
- DigitFrom (0) - начальное значение.
- DigitTo (10)- конечное значение.
- ScrollStep (0) - шаг курсора.
- accuracy (0.1)- точность измерений.
- LinearGap (4) - отступ линейки от полосы прокрутки.
- LinearsHeight (7) - размер больших штрихов линейки.
- LinearsUnderline (true) - подчеркивание линейки.
- ScrollerWidth (20) - ширина указателя.
- WidthSideRects (25) - ширина боковых прямоугольников.

- TextColor (000000) - цвет текста.
- BackColorUp (000000) - верхний цвет фона.
- BackColorDown (808080) - нижний цвет фона.
- SideRectsUpColor (f40000) - верхний цвет градиента боковых прямоугольников.
- SideRectsDownColor (7f0000) - нижний цвет градиента боковых прямоугольников.
- CursorUpColor (00f4ff) - верхний цвет градиента курсора
- CursorDownColor (007fff) - нижний цвет градиента курсора
- cursorform (Стрелка) - форма курсора:
  1. Стрелка.
  2. Эллипс.
  3. Прямоугольник.
  4. Треугольник.
- LinearVisible (true) - видимость линейки.
- linearorientation (left) - сторона отображения линейки:
  1. left - левая сторона.
  2. right - правая сторона.
- Enabled (true) - включение/отключение компонента.

## 8.2. Горизонтальная прокрутка

**Горизонтальная прокрутка** - компонент. Предназначен для выбора значения посредством перемещения указателя на шкале и последующей передачи значения в канал.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Вкл\Выкл - вывод, с которого идет управляющий сигнал.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>



➤ Частные свойства:

- TextSize (8) - размер текста.
- textstyle (FontStyleItalic) - стиль текста
- CurrentValue (0) - установка текущего значения.
- DigitFrom (0) - начальное значение.
- DigitTo (10) - конечное значение.
- ScrollStep (0) - шаг курсора.
- accuracy (0.1) - точность измерений.
- LinearGap (4) - отступ линейки от полосы прокрутки.
- LinearsHeight (7) - размер больших штрихов линейки.
- LinearsUnderline (true) - подчеркивание линейки.
- ScrollerWidth (20) - ширина указателя.
- WidthSideRects (25) - ширина боковых прямоугольников.
- CurrentValVisible (false) - отображение текущего значения.
- TextColor (000000) - цвет текста.
- BackColorUp (000000) - верхний цвет фона.
- BackColorDown (808080) - нижний цвет фона.
- SideRectsUpColor (f40000) - верхний цвет градиента боковых прямоугольников.
- SideRectsDownColor (7f0000) - нижний цвет градиента боковых  
прямоугольников.
- CursorUpColor (00f4ff) - верхний цвет градиента курсора
- CursorDownColor (007fff) - нижний цвет градиента курсора
- cursorform (Стрелка) - форма курсора:
  1. Стрелка.
  2. Эллипс.
  3. Прямоугольник.
  4. Треугольник.
- LinearVisible (true) - видимость линейки.
- linearorientation (Up) - сторона отображения линейки:
  1. Up - над линейкой.
  2. Down - под линейкой.
- Enabled (true) - включение/отключение компонента.
- scale (линейный) - масштаб шкалы линейный/логарифмический.



### **Пример**

**Пример** горизонтальной прокрутки: (находится в Примере компонента [Прослушивание канала](#)<sup>539)</sup>)

## 8.3. Демультимплексор

**Демультимплексор** - компонент. Предназначен для подключения единственного входного канала к одному из нескольких выходов, определяемых адресом.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Адрес - вывод, на который поступает адрес того выхода, к которому будет подключен вход.
- Вход - вывод, к которому подключается канал, который в дальнейшем будет коммутироваться.

#### ➤ Выходные:

- Выход 1 - вывод, к которому будет подключен канал входа, если будет выбран адрес Выхода 1.
- ...
- Выход N - вывод, к которому будет подключен канал входа, если будет выбран адрес Выхода N.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке](#).<sup>[92]</sup>

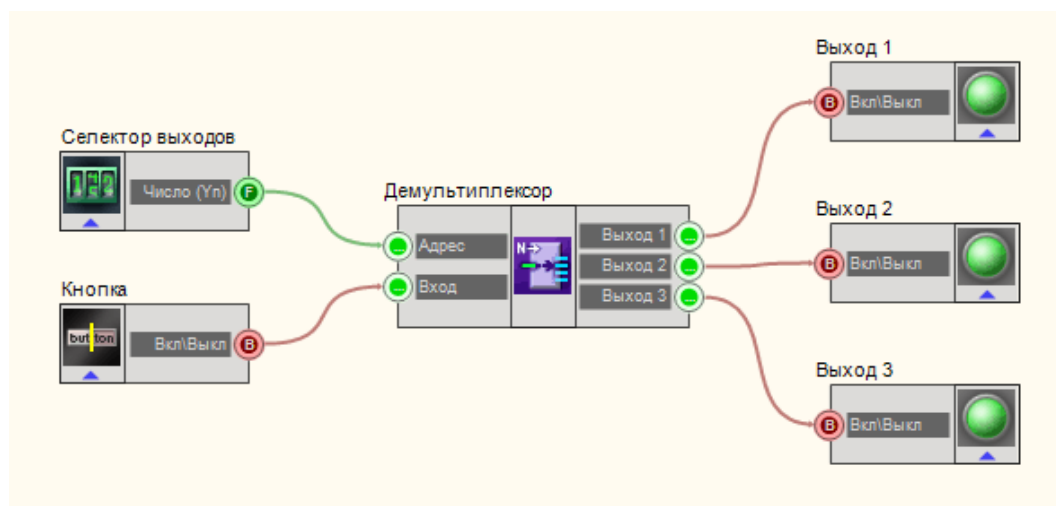
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Output (1) - задается количество выходов.
- State (1) - текущее подключение демультимплексора.



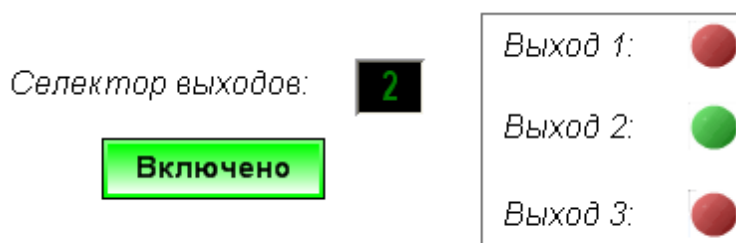
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Демultipлексор](#)<sup>[200]</sup> служат для подключения канала с [Кнопкой с фиксацией](#)<sup>[366]</sup> (в данном канале устанавливается высокий или низкий логический уровень в зависимости от состояния нажатия кнопки). [Селектор](#)<sup>[213]</sup> позволяет нам выбрать, к какому именно выходному каналу мы подключим канал с кнопкой.

### Результат работы проекта



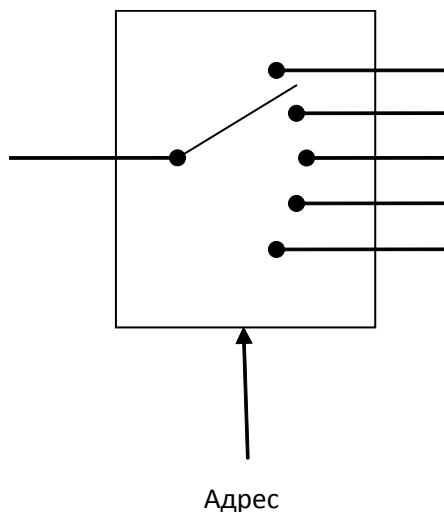
Данный пример непосредственно в ZETView



### Математическое описание

Демultipлексор – устройство, которое принимает входной сигнал и направляется им на один из нескольких выходов в соответствии с двоичным кодом, действующим на адресных входах. Остальные выходы в этом случае находятся либо в неактивном состоянии, либо в состоянии разомкнутой цепи. Аналогично работает дешифратор. Единственное отличие состоит в

том, что на входы подается только адрес, возбуждающий один из  $n$  возможных выходов.

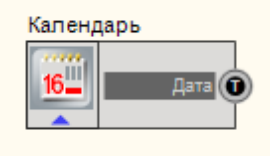
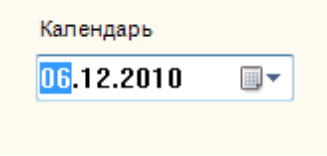


Дешифраторы обычно используются при сопряжении с микропроцессором, когда необходимо выполнить различные действия в зависимости от адреса. Другим применением общего использования дешифратора является организация (разрешение) последовательности действий, согласно достигнутого адреса, заданного выходом двоичного счетчика.

## 8.4.Календарь

**Календарь** - компонент. Предназначен для выбора и установки определенной даты и впоследствии ее использовании в проектах.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Дата - вывод, с которого идет текстовая строка с датой.

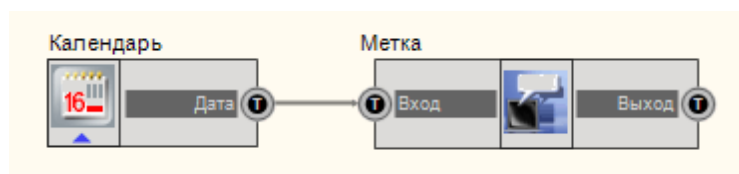
**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- dateFormat (ДД.ММ.ГГ)- установка текущей даты.

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Календарь](#)<sup>202</sup> служат для установки текущего значения даты. [Метка](#)<sup>554</sup> используется для дополнительного вывода на дисплей переданного значения даты.

**Результат работы проекта**

Дата

15.12.2010

Календарь

15.12.2010

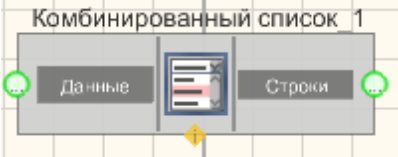
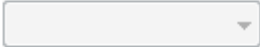


Данный пример непосредственно в ZETView

**8.5. Комбинированный список**

**Комбинированный список** - компонент. Предназначен для предоставления выбора потока данных путем выбора его в выпадающем списке.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Данные - строки, отображаемые в списке .

#### ➤ Выходные:

- Строки - вывод, с которого поступают данные, которые были выбраны в качестве одной из строк.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

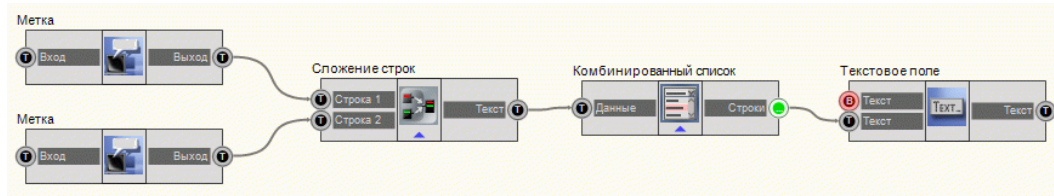
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Data - данные списка (записываются через разделитель "точка с запятой").
- Type (Строки) - выбор типа выходных значений:
  1. Строки.
  2. Численные значения.
  3. Индексы.
- CurrentPosition (0) - установка текущего элемента списка.



## Пример

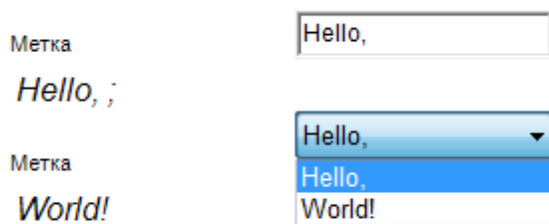
### Проект в SCADA ZETView



В этой схеме компонент [Комбинированный список](#)<sup>[203]</sup> служит для выборочной передачи текста с одной из меток в [Текстовое поле](#)<sup>[561]</sup>. Компонент [Сложение строк](#)<sup>[558]</sup> нужен для сцепления двух каналов с текстом в один общий. Необходимо обратить внимание, что в частных

свойствах [Сложении строк](#)<sup>[558]</sup> установлен символ-разделитель (в данном случае), который необходим, для передачи сцепленных строк в [Комбинированный список](#)<sup>[203]</sup> в виде 2 строк.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## 8.6.Константа

**Константа** - компонент. Предназначен для выбора и установки определенного постоянного значения, которое постоянно передается в канал.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Выходные:

- Константа (Yn) - вывод, с которого поступает установленное постоянное значение.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (0) - установка значения константы.

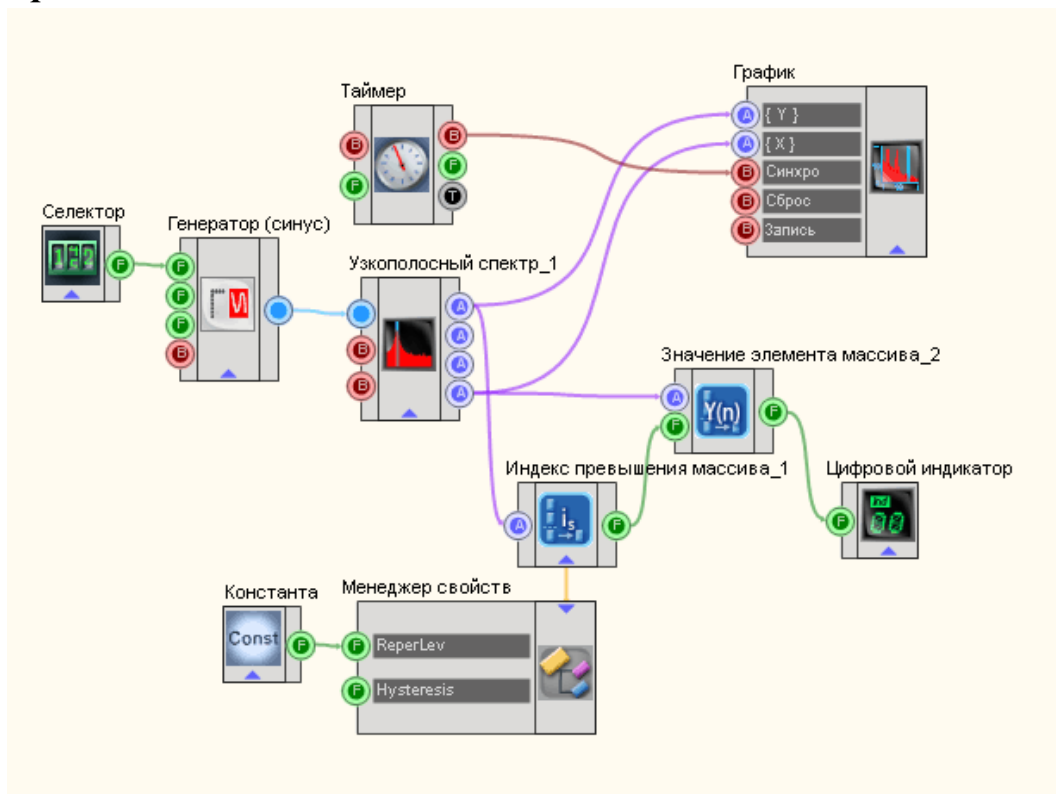


## Пример



Данный пример требует подключенного ZET-устройства, содержащего ЦАП.

### Проект в SCADA ZETView



В этой схеме компонент [Константа](#)<sup>[205]</sup> служит для установки значения уровня, с которым будут сравниваться все значения спектральной плотности сигнала на различных частотах. Если будет превышение этого уровня, то значение частоты, на которой это произошло, будет отображено на [Цифровом индикаторе](#)<sup>[362]</sup>. [Синусоидальный сигнал](#)<sup>[247]</sup> нужен для генерирования гармонического сигнала. [Узкополосный спектр](#)<sup>[128]</sup> используется для получения спектра сигнала. [График](#)<sup>[490]</sup> необходим для отображения зависимости спектральной плотности от частоты на дисплее. [Индекс превышения массива](#)<sup>[430]</sup> нужен для нахождения частоты, на которой будет превышено значение спектральной плотности, [Значение элемента массива](#)<sup>[428]</sup> необходим для того, чтобы получить конкретное значение частоты из массива, на котором совершится превышение.

### Результат работы проекта





Данный пример непосредственно в ZETView

## 8.7.Мультиплексор

**Мультиплексор** - компонент. Предназначен для подачи сигнала с одного из входов на выход, при этом нужный вход выбирается при помощи адреса.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Адрес - вывод, на который поступает адрес того входа, к которому будет подключен выход.
- Вход 1 - вывод, который будет подключен к каналу выхода, если будет выбран адрес Входа 1.
- ...
- Вход N - вывод, к которому будет подключен канал входа, если будет выбран адрес Входа N.

## ➤ Выходные:

- Выход - вывод, к которому подключается канал, который в дальнейшем будет коммутироваться.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

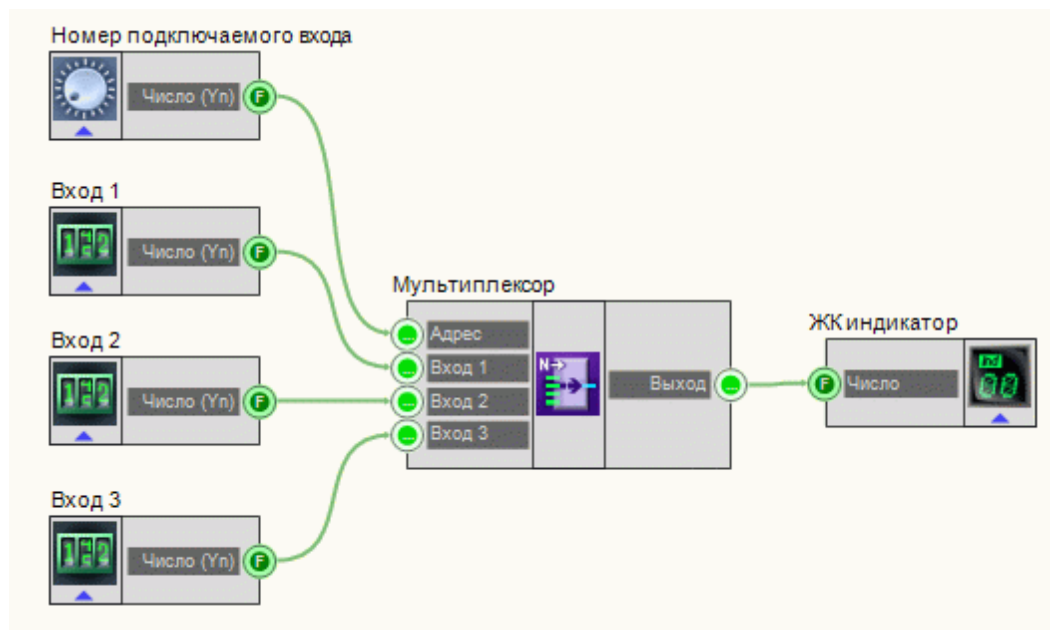
- Input (1) - задается количество входов.
- State (1) - текущее подключение мультиплексора.

## Примечание;

Передавать данные дальше мультиплексор будет только, если у него будет заполнено поле "m\_sHelpString". Без указания "m\_sHelpString" компонент "График" не будет отображать массив, прошедший через мультиплексор.

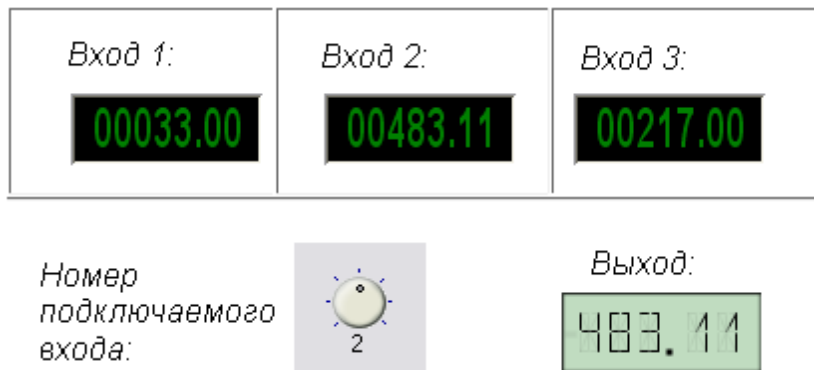
---

***Пример******Проект в SCADA ZETView***



В этой схеме компонент [Мультимплексор](#)<sup>[207]</sup> служит для подключения канала с Кнопкой (в данном канале устанавливается высокий или низкий логический уровень в зависимости от состояния нажатия кнопки). [Селектор](#)<sup>[215]</sup> позволяет нам выбрать к какому, именно выходному каналу мы подключим канал с кнопкой.

### Результат работы проекта

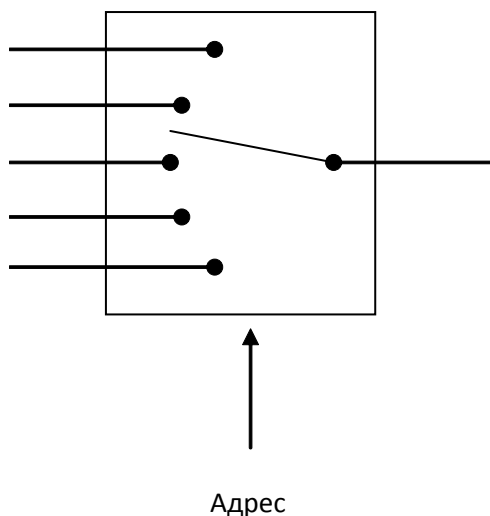


Данный пример непосредственно в ZETView



### *Математическое описание*

Мультиплексор – устройство, которое подключает один из  $n$  входных сигналов к единственной выходной линии. Выбор подключаемого входа осуществляется при помощи адреса, передаваемого по специальным линиям. Мультиплексоры иногда называют селекторами данных.



Если сравнивать принципы работы аналоговых и цифровых мультиплексоров, то они коренным образом отличны друг от друга. В аналоговых образуется электрический контакт между входом и выходом. В цифровых же значение входного сигнала копируется на выход схемы.

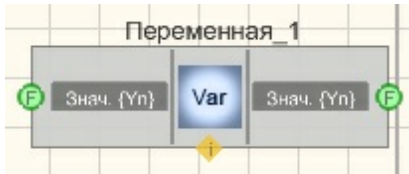
Структурно мультиплексоры состоят из коммутатора, который обеспечивает соединение входов с одним выходом. Коммутатор управляется при помощи адресных и иногда разрешающих (стробирующих) входов.

В цифровых мультиплексорах логические элементы коммутатора и дешифратора обычно объединяются.

## 8.8.Переменная

**Переменная** - компонент. Предназначен для передачи на выход переменного значения.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входы:

- Значение (Yn) - входное значение.

## ➤ Выходы:

- Значение (Yn) - выходное значение.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

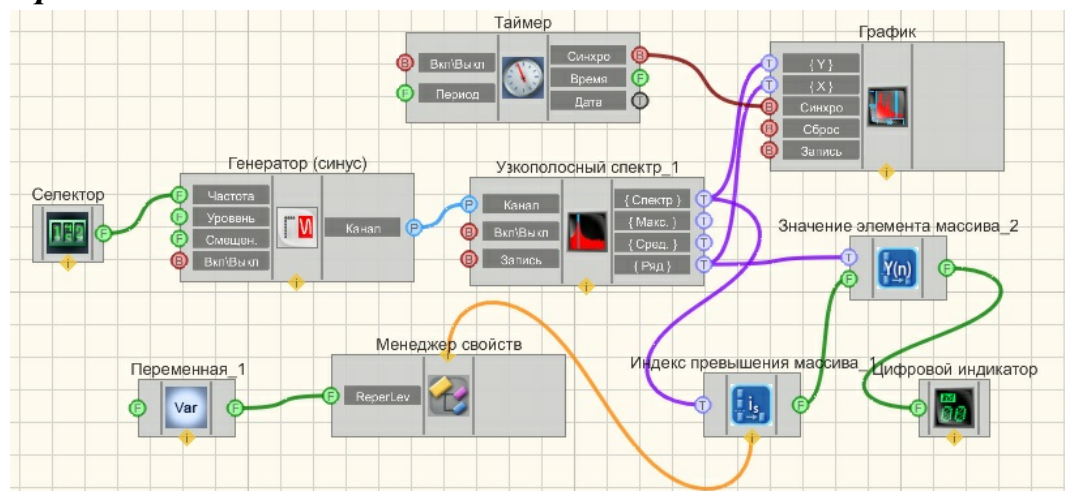
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (0) - установка текущего значения.

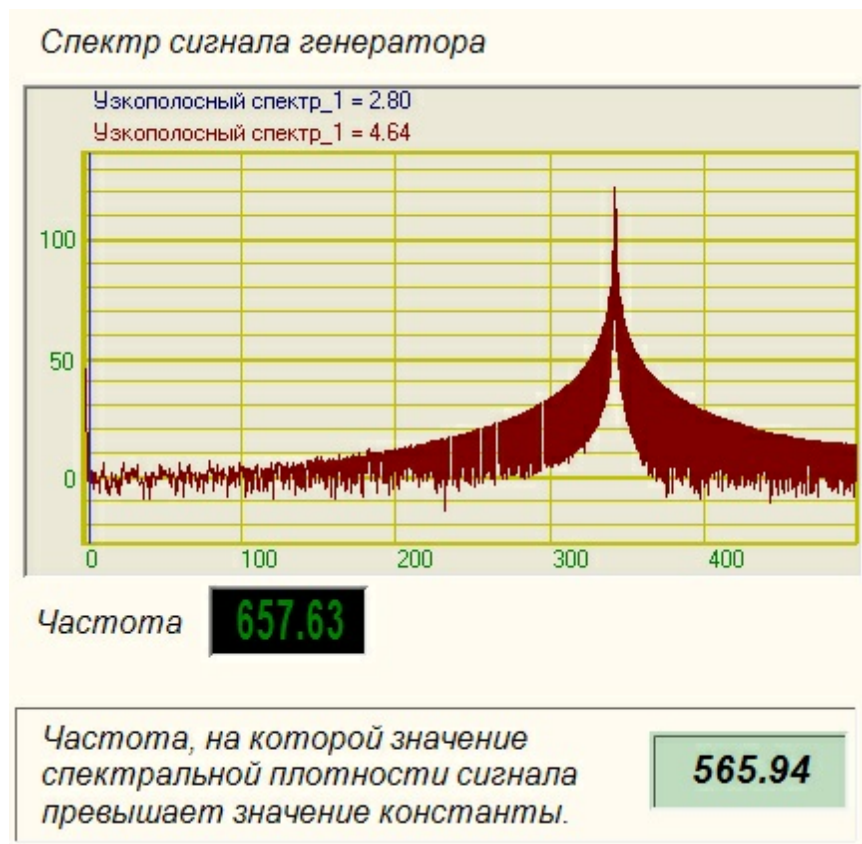
**Пример**

Данный пример требует подключенного ZET-устройства, содержащего ЦАП.

**Проект в SCADA ZETView**

В этой схеме компонент [Переменная](#)<sup>[210]</sup> служит для установки значения уровня, с которым будет сравниваться все значения спектральной плотности сигнала на различных частотах. Если будет превышение этого уровня, то значение частоты, на которой это произошло, будет отображено на [Цифровом индикаторе](#)<sup>[362]</sup>. [Синусоидальный сигнал](#)<sup>[247]</sup> нужен для генерирования гармонического сигнала. [Узкополосный спектр](#)<sup>[128]</sup> используется для получения спектра сигнала. [График](#)<sup>[490]</sup> необходим для отображения зависимости спектральной плотности от частоты на дисплее. [Индекс превышения массива](#)<sup>[430]</sup> нужен для нахождения частоты, на которой будет превышено значение спектральной плотности, [Значение элемента массива](#)<sup>[428]</sup> необходим для того, чтобы получить конкретное значение частоты из массива, на котором совершится превышение.

### Результат работы проекта

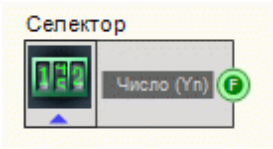
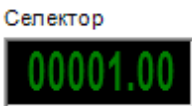


Данный пример непосредственно в ZETView

## 8.9.Селектор № 1

**Селектор №1** - компонент. Предназначен для выбора численных значений и передачи их в канал.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

Не имеет.

#### ➤ Выходные:

- Число (Yn) - выходной контакт, с которого поступает число, введенное пользователем на селекторе.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (1) - текущее значение, выставленное на селекторе, сохраняется после закрытия проекта. ZETView оповещает пользователя, если параметры изменены, и предлагает сохранить данные или отменить сохранение. Если сохранить проект, то сохраняются также текущие значения на селекторах.
- Min (0) - минимально возможное устанавливаемое значение.
- Max (1000.000000) - максимально возможное устанавливаемое значение.
- Color (008000) - цвет текста. По умолчанию зеленый.
- FonColor (000000) - цвет фона. По умолчанию черный.
- Status (true) - активация/деактивация селектора.
- NumCount (5) - количество отображаемых целых знаков. По умолчанию 5. Каким бы не было значение параметров Min и Max, если количество знаков недостаточно, пользователь сможет ввести число максимально близкое к указанному, но ограниченное количеством знаков. Например, если Max=1000, а NumCount=3, максимальное число, которое можно ввести с такого селектора 999.
- NumDivCount (2) - количество отображаемых десятичных знаков после запятой.
- Activate (true) - Состояние работы. Если значение = false, пользователь не сможет взаимодействовать с селектором.

**Описание:**

Чтобы начать работу с селектором нужно запустить проект на исполнение. Для ввода данных с селектора нужно щелкнуть 1 раз левой кнопкой мыши по нужному разряду и покрутить колесико мыши.

Второй метод использования данного компонента:

- 1) Двойной клик по компоненту, появится окно ввода значения:

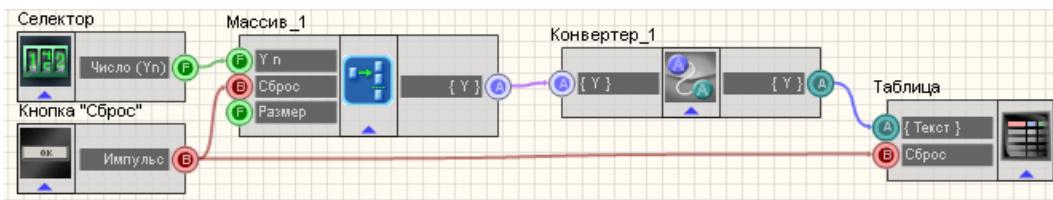


- 2) Далее вводится значение и нажимается кнопка Enter

Второй метод использования компонента "Селектор" является более правильным. Во-первых, подобным образом проще ввести необходимое значение. Во-вторых, если необходимо подать с селектора 2 одинаковых значения подряд, нужно использовать данный метод, так как селектор передает данные в канал, лишь при изменении значения. То есть, если на селекторе выставлено число "1", он передаст его канал только один раз, чтобы компонент снова передал значение, значение нужно изменить.

Для ввода двух одинаковых значений подряд достаточно сделать двойной клик по селектору и нажать Enter. Тогда селектор снова передаст последнее значение. (См Пример1). Подобный метод ввода бывает необходим при работе с некоторыми компонентами, например, такими как [Ячейка Excel](#)<sup>[620]</sup>.

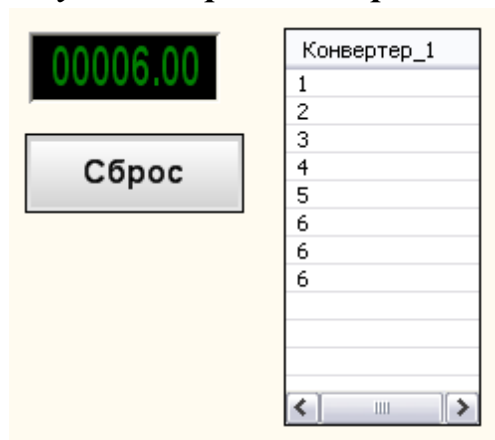
Селектор является очень распространенным и часто используемым компонентом в ZETView. Используется как универсальное средство ввода чисел.

**Пример****Пример 1****Проект в SCADA ZETView**



В данном примере пользователем вводятся числа с селектора. После чего поступают на компонент [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup>. Далее числовой массив поступает на [Конвертер \(стр.мас.-числ.мас.\)](#)<sup>[373]</sup>, где преобразуется в строковый массив и выводится в [Таблицу данных](#)<sup>[496]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс" очищает таблицу и сбрасывает значения на компоненте [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup>.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

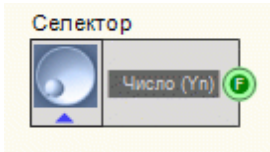

### *Пример 2*

*Пример* создания генератора линейно-частотно модулированного сигнала входа: (находится в Примере компонента [Генератор \(ЛинЧМ\)](#)<sup>[231]</sup>)

## 8.10.Селектор № 2

**Селектор №2** - компонент. Предназначен для выбора численных значений и передачи их в канал.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Число (Yn) - вывод, на котором выставляется значение, которое задаст пользователь.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (1) - текущее значение.
- Status (true) - активация/деактивация селектора.
- Min (0) - минимально возможное устанавливаемое значение.
- Max (9) - максимально возможное устанавливаемое значение.
- Fix (false) - установка режима переключения:
  1. False - позволяет переключаться по всему диапазону шкалы.
  2. True - позволяет переключаться только между целыми значениями.
- FontName (Arial) - шрифт чисел на шкале.
- FontSize (14) - размер шрифта.
- Mark (10) - количество основных делений шкалы.
- Line (5) - количество дополнительных делений шкалы.
- Point (false) - установка дополнительных промежуточных меток.
- Unit (c) - установка единиц измерения.
- FonColor (f0f0f0) - установка цвет фона.
- Color (cccccc) - цвет внутренней области шкалы.



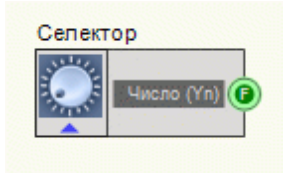
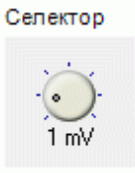
### **Пример**

**Пример** создания генератора прямоугольных импульсов: (находится в Примере компонента [Генератор импульсов](#)<sup>[229]</sup>)

## 8.11.Селектор № 3

**Селектор №3** - компонент. Предназначен для выбора численных значений и передачи их в канал. Селектор исполнен в виде крутящегося регулятора. Значения, которые можно выбрать на данном селекторе определяются его настройками: количество делений и шаг.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤Выходные:

- Число (Yn) - вывод, на котором выставляется значение, которое задаст пользователь.

### Настраиваемые свойства:

#### ➤Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤Частные свойства (в скобках значение, установленное по умолчанию):

- Min (0) - минимально возможное устанавливаемое значение.
- Max (10) - максимально возможное устанавливаемое значение.
- Status (true) - активация/деактивация селектора.
- Position (1) - текущее значение селектора.
- UnitName (mV) - единица измерения.
- Scale (1) - множитель значений. Фактически определяет шаг между делениями шкалы.
- BackColor (f0f0f0) - цвет фона шкалы.



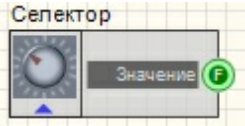
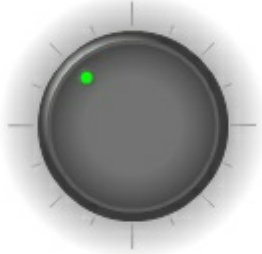
### **Пример**

**Пример** создания генератора пилообразного сигнала: (находится в Примере компонента [Пилообразный сигнал](#)<sup>[239]</sup>)

## 8.12.Селектор № 4

**Селектор №4** - компонент. Предназначен для выбора численных значений. Селектор исполнен в виде бесконечного крутящегося регулятора. Значения, которые можно выбрать на данном селекторе определяются его настройками: свойства цветов, изменение на один оборот, текущее значение, дискретность поворота.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Значение - вывод, на котором выставляется управляющий сигнал, которое задаст пользователь.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

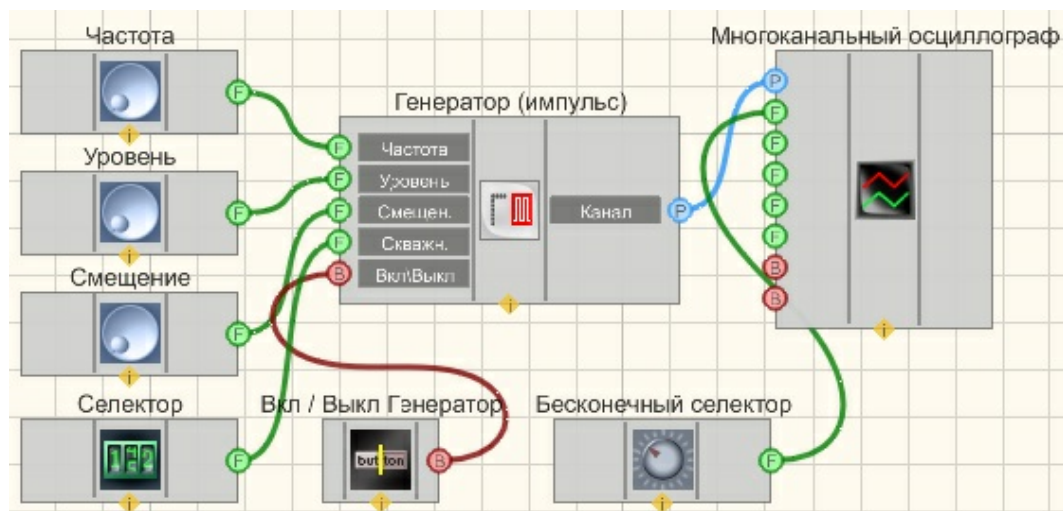
- GradientBackColor (373737)- устанавливается цвет селектора.
- selectorform (Окружность) - определяется форма селектора:
  1. Цилиндр.
  2. Окружность.
  3. Сфера.
- cursorform (Окружность) - определяется форма курсора:
  1. Треугольник.
  2. Окружность.
- CursorColor (005eff)- определяется цвет курсора (для стиля "С заливкой"). По умолчанию синий.
- cursorstyle (Градиент) - устанавливается стиль курсора:
  1. С заливкой
  2. Градиент
- shade (Темная) - устанавливается тень:

1. Нет.
  2. Темная.
  3. Светлая.
- LinearVisible (True) - устанавливается видимость линейки (True/False).
  - Linearstyle (Разной длины) - определяется стиль линейки:
    1. Одной длины
    2. Разной длины
  - LinearColor (a0a0a4)- устанавливается цвет линейки.
  - ValueOnLap (10) - определяется изменение значения при повороте на 360 градусов.
  - CurrentValue (0) - устанавливается текущее значение.
  - StepMove (false) - определяется движение по шагам (Вкл./Выкл.).
  - ValueInStep (0.5) - устанавливается дискретность шага (не может быть больше, чем ValueOnLap / 8).



## Пример

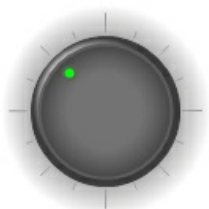
### Проект в SCADA ZETView



В этой схеме компонент [Пилообразный сигнал](#)<sup>[572]</sup> создает пилообразный сигнал, с параметрами, которые задаются при помощи [Селекторов](#)<sup>[213]</sup>. Для просмотра сгенерированного сигнала служит компонент [Многоканальный осциллограф](#)<sup>[494]</sup>. Дополнительный [Селектор](#)<sup>[217]</sup> позволяет устанавливать тип пилы (нарастающий, ниспадающий, треугольный). [Бесконечный селектор](#)<sup>[218]</sup> позволяет изменять интервал на многоканальном осциллографе. Результат работы составленной программы можно увидеть в интерфейсе оператора.

## Результат работы проекта

Длительность:



Частота:



Уровень:



Смещение:



Скважность:

0.49

Включено

Данный пример непосредственно в ZETView

### 8.13.Список

**Список** - компонент. Предназначен для предоставления выбора потока данных путем выбора его в выпадающем списке.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора

**Параметры:**

➤ Входные:

- Данные - вывод, к которому подключается канал с данными, которые требуется представить в виде списка с возможностью выбора.

➤ Выходные:

- Строки - вывод, с которого поступают данные, которые были выбраны в качестве одной из строк.

**Настраиваемые свойства:**

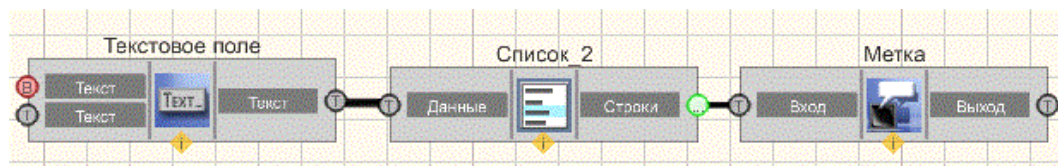
➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):
- Data - данные списка.
- Type (Строки)- выбор типа списка:
  1. Строки.
  2. Численные значения.
  3. Индексы.



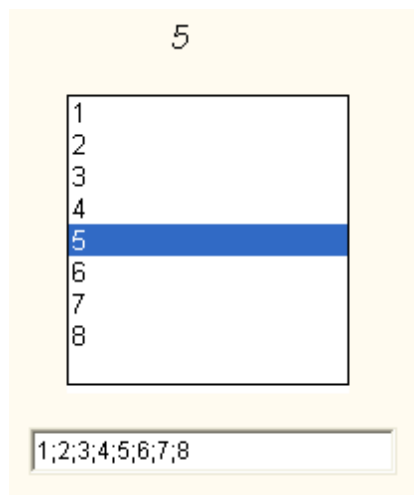
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Список](#)<sup>[220]</sup> служит для выборочной передачи информации с [Текстового поля](#)<sup>[561]</sup>. Компонент [Метка](#)<sup>[554]</sup> нужен для вывода выбранной части сообщения на дисплей.

### Результат работы проекта



Данный пример непосредственно в ZETView

## Глава 9. Генератор (ЦАП)

### 9.1. Амплитудная модуляция

**Амплитудная модуляция** - компонент. Позволяет генерировать сигнал, создаваемый по специальному закону модуляции частоты, при которой изменяемым параметром несущего сигнала является его амплитуда.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Несущая частота - параметр, который служит для задания несущей частоты, Гц.
- Модуляционная частота - параметр, который служит для задания конечной частоты частотного диапазона, в котором будет генерироваться белый шум, Гц.
- Уровень - параметр, который задает амплитуду выходного сигнала, В.
- Глубина - параметр, задающий коэффициент амплитудной модуляции.
- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.

##### ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

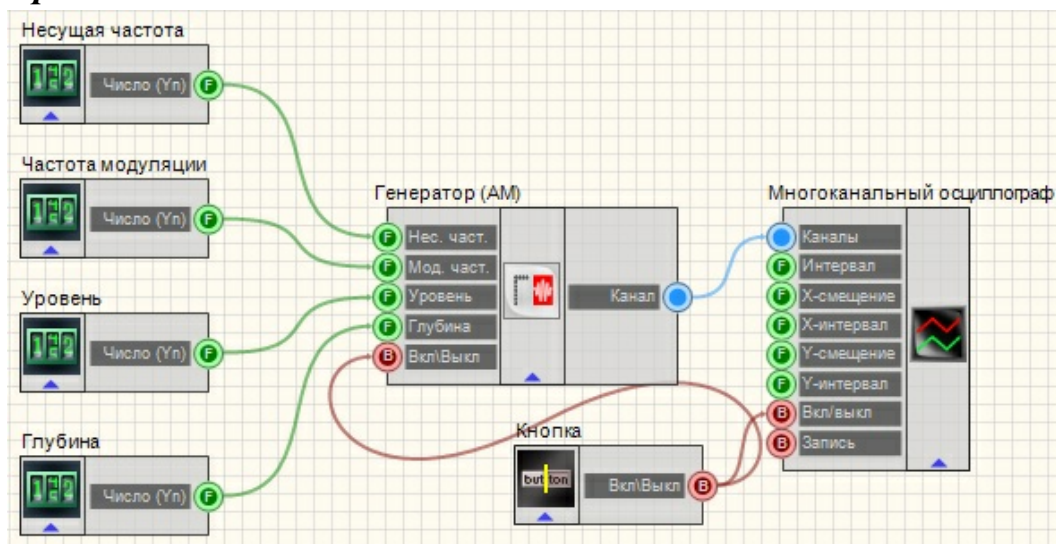
- CentralFrequency (1000) - частота несущего сигнала, Гц.
- ModFrequency (1) - частота амплитудной модуляции, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Depth (0) - глубина амплитудной модуляции.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.





## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Амплитудная модуляция](#)<sup>[222]</sup> на несущей частоте производит амплитудное модулирование сигнала. [Селекторы](#)<sup>[213]</sup> служат для задания несущей частоты, уровня, глубины и частоты амплитудной модуляции. [Кнопка с фиксацией](#) служит для включения или отключения.

### Результат работы проекта



Данный пример непосредственно в ZETView



### *Математическое описание*

Амплитудная модуляция – вид модуляции, при которой изменяемым параметром несущего сигнала является его амплитуда.

Пусть  $S(t)$  – информационный сигнал,  $|S(t)| < 1$ .

$U_c(t)$  несущее колебание.

Тогда амплитудно-модулированный сигнал  $U_{am}$  может быть записан следующим образом:

$$U_{am}(t) = U_c(t)[1 + mS(t)]. \quad (I)$$

Здесь  $m$  – некоторая константа, называемая коэффициентом модуляции. Формула (I) описывает несущий сигнал  $U_c$ , модулированный по амплитуде сигналом  $S(t)$  с коэффициентом модуляции  $m$ . Предполагается также, что выполнены условия:

$$|S(t)| < 1; \quad 0 < m \leq 1. \quad (II)$$

Выполнение условий (II) необходимо для того, чтобы выражение в квадратных скобках в (I) всегда было положительным. Если оно может принимать отрицательные значения в какой-то момент времени, то происходит так называемая перемодуляция (избыточная модуляция).

Допустим, мы хотим промодулировать несущее колебание моногармоническим сигналом. Выражение для несущего колебания с частотой  $\omega_c$ , начальную фазу положим равной нулю, имеет вид:

$$U_c(t) = C \sin(\omega_c t)$$

Выражение для синусоидального сигнала с частотой  $\omega_s$  (сигнал подлежит передаче) имеет вид:

$$U_s(t) = U_0 \sin(\omega_s t + \varphi),$$

где  $\varphi$  – начальная фаза. Тогда в соответствии с (I)

$$U_{am}(t) = C[1 + mU_0 \sin(\omega_s + \varphi)] \sin(\omega_c t)$$

Приведенная выше формула для  $y(t)$  может быть записана в следующем виде:

$$U_{am}(t) = C \sin(\omega_c t) + \frac{mCU_0}{2} (\cos((\omega_c - \omega_s)t - \varphi) - \cos((\omega_c + \omega_s)t + \varphi))$$

## 9.2. Воспроизведение из файла

**Воспроизведение из файла** - компонент. Позволяет генерировать сигнал по взятым из файла параметрам.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входящие:

- Файл - имя файла, из которого будут браться данные сигнала для воспроизведения.
- Усиление - коэффициент усиления сигнала из файла.
- Цикличность - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.

➤ Выходные

- Канал - вывод уже сгенерированного по файлу сигнала.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

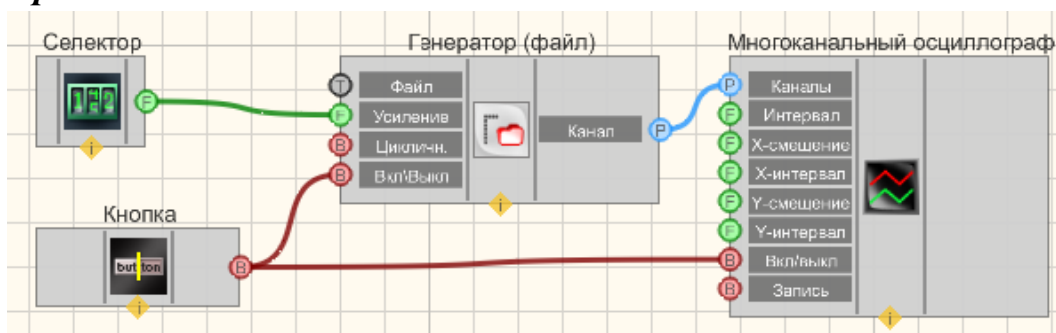
- FileName - выбор необходимого файла с заранее в него записанным сигналом.

- Amplify (1) - коэффициент усиления сигнала из файла.
- Cycle (true) - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.



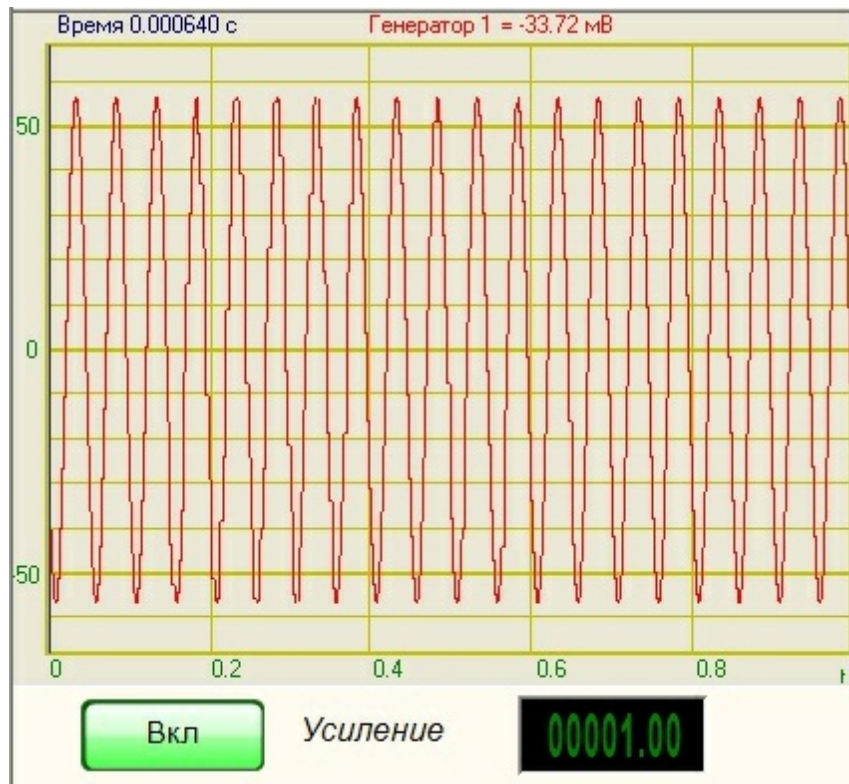
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Воспроизведение из файла](#)<sup>[225]</sup> выдает записанный ранее сигнал из файла на вход [Многоканального осциллографа](#)<sup>[494]</sup>. При помощи [Селектора](#)<sup>[213]</sup> можно задавать коэффициент усиления для воспроизводимого из файла сигнала. [Кнопка с фиксацией](#) служит для включения или отключения.

### Результат работы проекта



Данный пример непосредственно в ZETView

### 9.3. Входной канал

**Входной канал** - компонент. Позволяет выбирать один из каналов подключенного физического ZET-устройства. Передает на выход ID выбранного канала следующему компоненту, который, в свою очередь, принимает сигналы ZET-устройства и производит их обработку.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора

**Параметры:**

➤ Входные:

- Канал - входной канал.

- Коэфф. - коэффициент усиления/ослабления сигнала.
- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - выходной канал генератора.

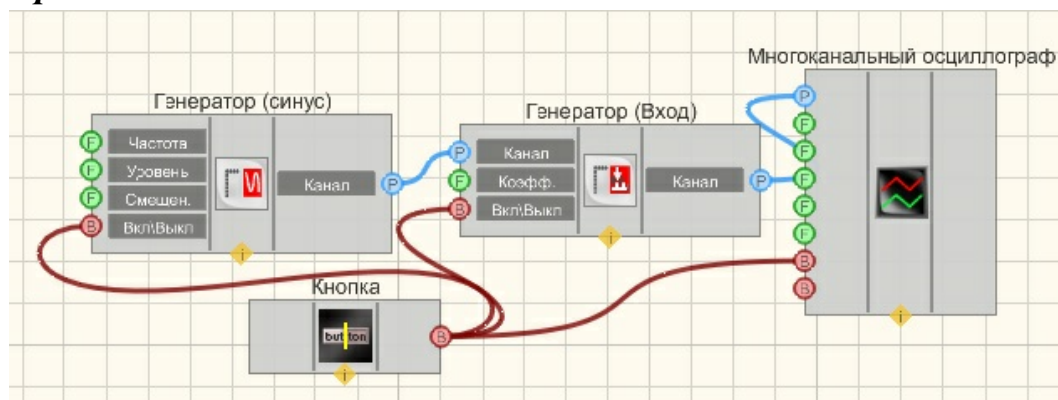
#### Настраиваемые свойства:

- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>[92]</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):
  - Коэффициент (1) - коэффициент усиления/ослабления сигнала.
  - Activate (false) - состояние подачи сигнала (подан или отключен).
  - Number (0) - номер генератора в системе, начиная с нулевого.
  -



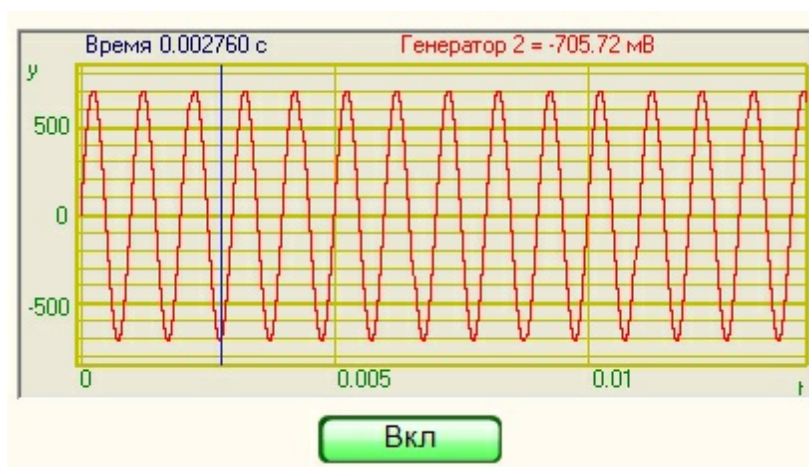
### Пример

#### Проект в SCADA ZETView



В этой схеме компонент [Входной канал](#)<sup>[227]</sup> получает сигнал с [Генератора синусоидального сигнала](#)<sup>[247]</sup>, а затем выводит на вход [Многоканального осциллографа](#)<sup>[494]</sup>. При помощи [Кнопка с фиксацией](#)<sup>[366]</sup> можно включать и выключать сигнал. [Кнопка с фиксацией](#) служит для включения или отключения.

#### Результат работы проекта



Данный пример непосредственно в ZETView

## 9.4. Импульсный сигнал

**Импульсный сигнал** - компонент. Позволяет генерировать прямоугольные импульсы с заданной частотой, скважностью, уровнем и постоянной составляющей.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Вводные:

- Частота - параметр, который задает значение частоты импульсов, Гц.
- Уровень - параметр, предназначенный для задания среднеквадратичного значения (СКЗ) уровня, с которым будут генерироваться ЛогЧМ, В.
- Смещение - параметр, позволяющий задавать смещение постоянной составляющей, В.
- Скважность - (коэффициент заполнения) предназначен для задания коэффициента заполнения (отношение длительности импульса к периоду следования), с которым будет генерироваться сигнал.



- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

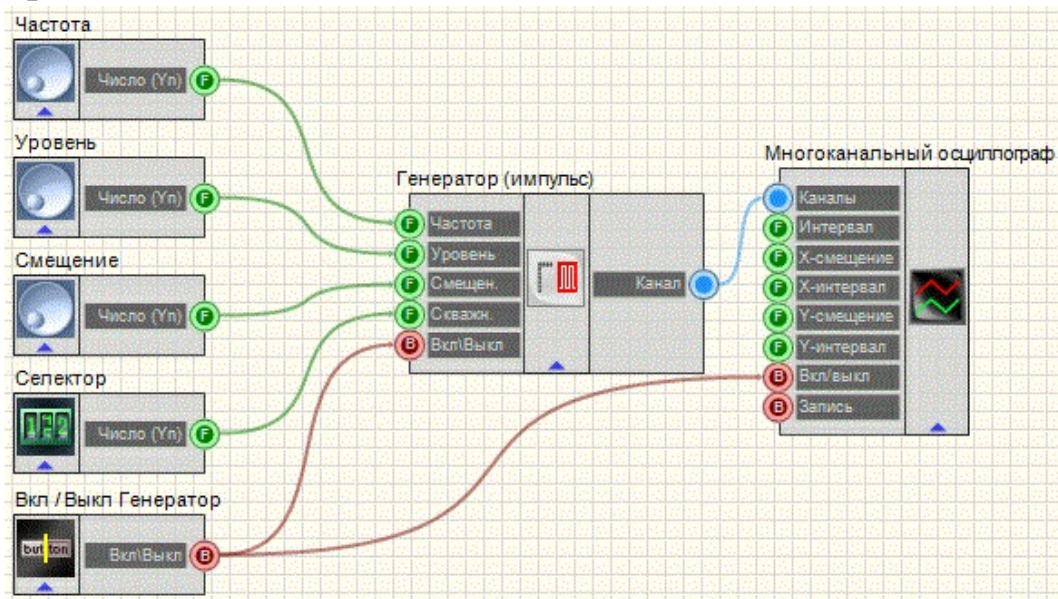
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Frequency (100) - частота выходного сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Shift (0) - смещение постоянной составляющей, В.
- Porosity (0.5) - параметр, отвечающий за скважность импульсов (отношение длительности импульса к периоду следования).
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.



## Пример

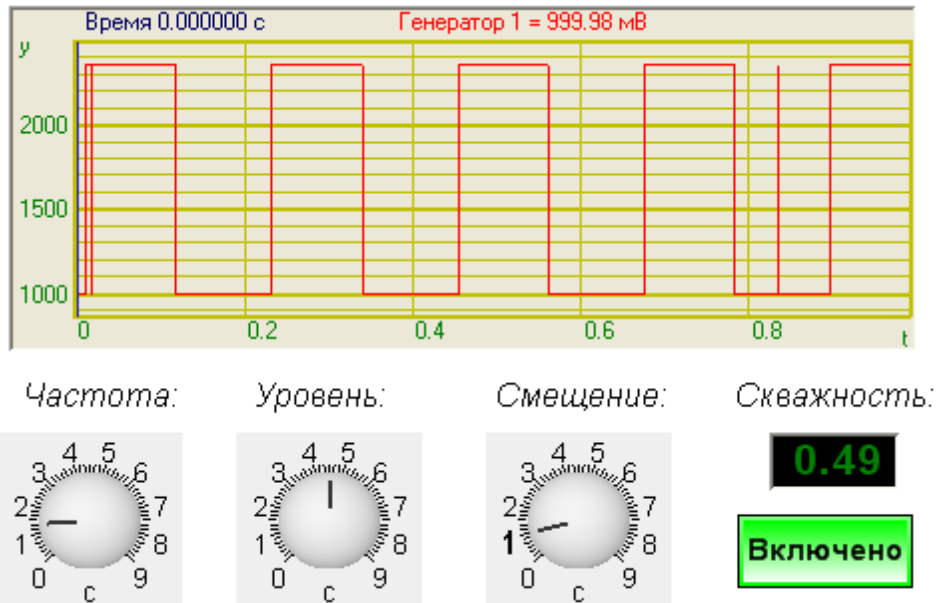
### Проект в SCADA ZETView



В этой схеме компонент [Импульсный сигнал](#)<sup>[229]</sup> генерирует прямоугольные импульсы. [Селекторы](#)<sup>[213]</sup> (а также [другой тип](#)<sup>[215]</sup>) служат для задания частоты, уровня, смещения, скважности. [Кнопка](#)<sup>[366]</sup> необходима для включения/выключения [Импульсного сигнала](#)<sup>[229]</sup>. [Кнопка с фиксацией](#)<sup>[229]</sup> служит для включения или отключения.



## Результат работы проекта



Данный пример непосредственно в ZETView

## 9.5. Линейная частотная модуляция

**Линейная частотная модуляция** - компонент. Позволяет генерировать сигнал, создаваемый по специальному закону модуляции частоты, при которой частота колебаний изменяется по линейному закону в некотором заданном диапазоне циклически.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
<p>Линейная частотная модуляция 1</p> <p>Н. част. К. част. Уровень Скорость Циклич. Вкл/Выкл</p> <p>Канал</p> <p>The design mode view shows a control panel with six parameters: Initial frequency (Н. част.), Final frequency (К. част.), Level (Уровень), Speed (Скорость), Cyclical (Циклич.), and On/Off (Вкл/Выкл). A 'Канал' (Channel) button is also present.</p>	Не имеет

### Параметры:

#### ➤ Входные:

- Начальная частота - предназначено для задания начальной частоты частотного

диапазона, в котором будет генерироваться ЛинЧМ, Гц.

- Конечная частота - параметр, предназначенный для задания конечной частоты частотного диапазона, в котором будет генерироваться ЛинЧМ, Гц.
  - Уровень - параметр, предназначенный для задания среднеквадратичного значения (СКЗ) уровня, с которым будут генерироваться ЛинЧМ, В.
  - Скорость - параметр, предназначенный для задания скорости изменения частоты с которой будет генерироваться ЛинЧМ, Гц/с.
  - Цикличность - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
  - Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) 

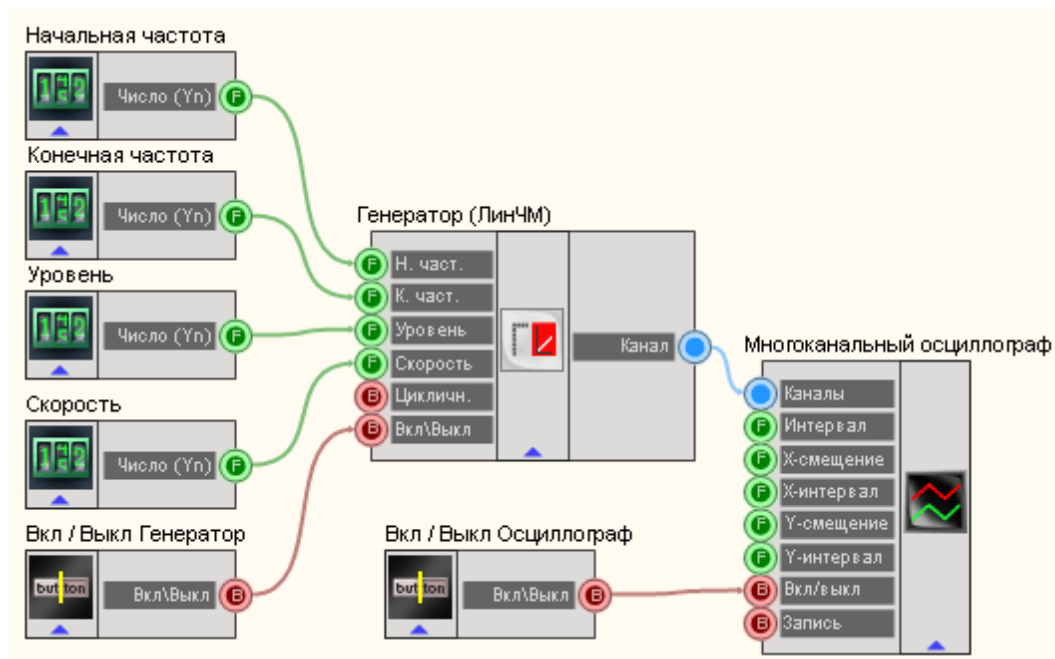
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- StartFrequency (0) - установка начальной частоты сигнала, Гц.
- EndFrequency (1000) - конечная частота сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Speed (1) - параметр, предназначенный для задания скорости изменения частоты с которой будет генерироваться ЛинЧМ, Гц/с.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Cyclic (true) - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Number (0) - номер генератора в системе, начиная с нулевого.



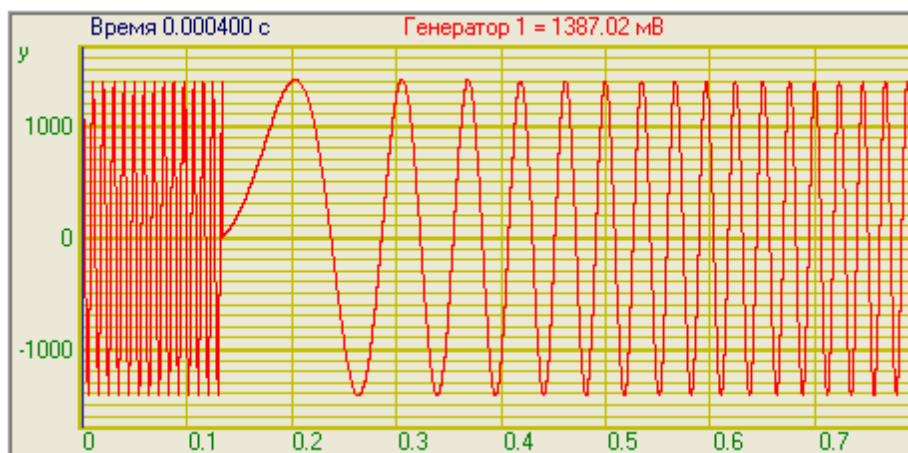
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Линейная частотная модуляция](#)<sup>[231]</sup> генерирует линейно-частотно-модулированный сигнал, параметры которого задаются при помощи [Селекторов](#)<sup>[213]</sup> (начальная частота, конечная частота, уровень, скорость). [Кнопки](#)<sup>[366]</sup> позволяют включать и отключать [Синусоидальный сигнал](#)<sup>[247]</sup> и [Многоканальный осциллограф](#)<sup>[494]</sup>. Осциллограф служит для просмотра конечного вида сигнала. [Кнопка с фиксацией](#) служит для включения или отключения.

### *Результат работы проекта*



Начальная частота:

00001.00

Осциллограф:

Конечная частота:

00110.00

Включено

Скорость:

00073.00

Генератор:

Уровень:

00001.00

Включено

Данный пример непосредственно в ZETView



### Математическое описание

Изменение частоты  $f(t)$  внутри импульсов с ЛЧМ происходит по линейному закону.

$$f(t) = f_0 + bt, -\frac{T_c}{2} \leq t \leq \frac{T_c}{2}$$

Где  $f_0 = \frac{(F_{max} + F_{min})}{2}$  - центральное значение несущей частоты.  $b = \frac{F_{max} - F_{min}}{T_c}$  - база (крутизна изменения частоты) ЛЧМ сигнала.  $T_c$  - длительность сигнала.

$F_{max}$ ,  $F_{min}$  - максимальное и минимальное значение частоты радиосигнала.

Фаза сигнала с ЛЧМ определяется как:

$$\varphi(t) = 2\pi \int_0^t f(t) dt = 2\pi(f_0 t + \frac{b}{2} t^2)$$

Тогда ЛЧМ сигнал может быть описан выражением:

$$S_{\text{LФМ}}(t) = S_0 \cos\{\varphi_0 + \varphi(t)\} = S_0 \cos\{\varphi_0 + 2\pi(f_0 t + \frac{b}{2} t^2)\}$$

Или в комплексном виде:

$$S_{\text{LФМ}}(t) = S_0 e^{j\{\varphi_0 + 2\pi(f_0 t + \frac{b}{2} t^2)\}}$$

Где  $S_0$  - амплитуда сигнала;  $j$  – мнимая единица;  $\varphi_0$  – начальная фаза.

ЛЧМ-сигналы применяются в радиолокации в качестве способа формирования и обработки зондирующего импульса. Применение ЛЧМ-сигнала позволяет повысить точность измерений в радиолокации. Также данный тип сигналов нашел место в обработке изображений, где имеет место перспектива (явление кажущегося искажения пропорций и формы тел при их визуальном наблюдении. Например, два параллельных рельса кажутся сходящимися в точку на горизонте.). В трехмерной радиотомографии, где использование ЛинЧМ сигнала позволяет проводить дистанционное бесконтактное исследование скрытых объектов и их внутренней структуры.

## 9.6. Логарифмическая частотная модуляция

**Логарифмическая частотная модуляция** - компонент. Позволяет генерировать сигнал, создаваемый по специальному закону модуляции частоты, при которой частота колебаний изменяется по логарифмическому закону в некотором заданном диапазоне циклически.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Начальная частота - предназначено для задания начальной частоты частотного диапазона, в котором будет генерироваться ЛогЧМ, Гц.
- Конечная частота - параметр, предназначенный для задания конечной частоты частотного диапазона, в котором будет генерироваться ЛогЧМ, Гц.
- Уровень - параметр, предназначенный для задания среднеквадратичного значения (СКЗ) уровня, с которым будут генерироваться ЛогЧМ, В.
- Скорость - параметр, предназначенный для задания скорости изменения частоты относительно октав в секунду, с которой будет генерироваться ЛогЧМ.
- Цикличность - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.

#### ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

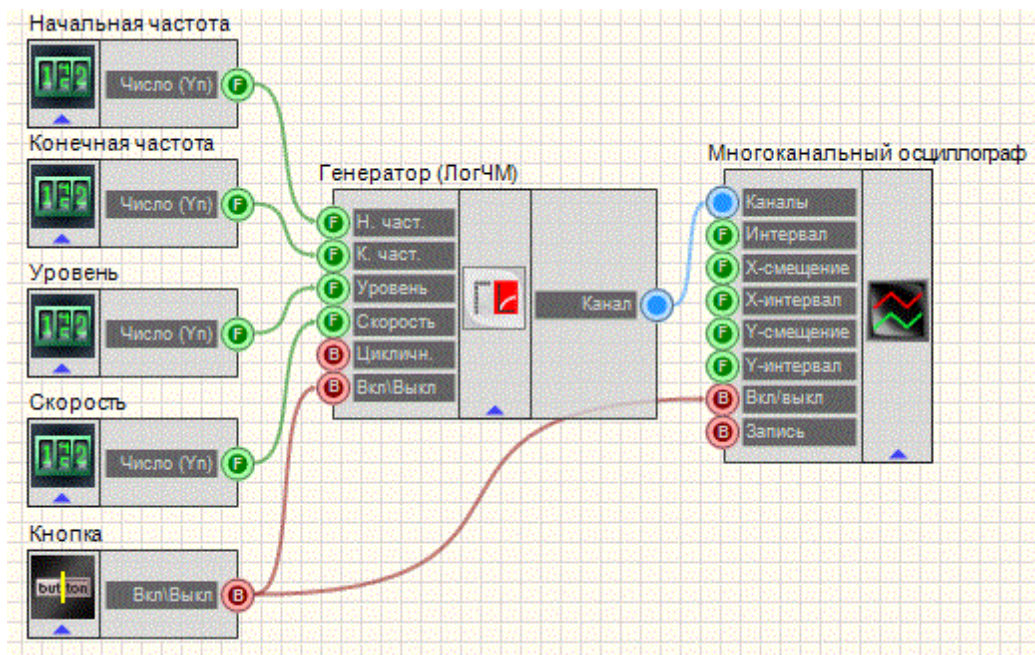
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- StartFrequency (0) - установка начальной частоты сигнала, Гц.
- EndFrequency (1000) - конечная частота сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Speed (1) - параметр, предназначенный для задания скорости изменения частоты относительно октав в секунду, с которой будет генерироваться ЛогЧМ.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Cyclic (true) - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Number (0) - номер генератора в системе (начиная с нулевого).



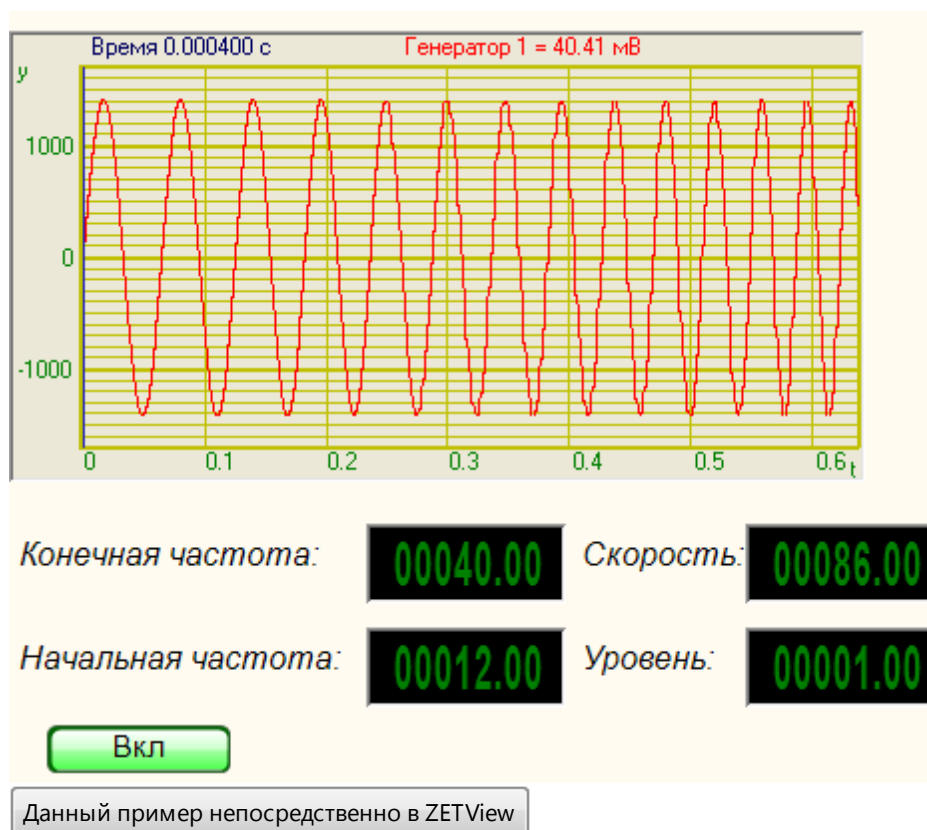
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Логарифмическая частотная модуляция](#)<sup>[235]</sup> генерирует логарифмически-частотно-модулированный сигнал, параметры которого задаются при помощи [Селекторов](#)<sup>[213]</sup> (начальная частота, конечная частота, уровень, скорость). [Многоканальный осциллограф](#)<sup>[494]</sup> используется для графического отображения сигнала. [Кнопка с фиксацией](#) служит для включения или отключения.

### Результат работы проекта



### Математическое описание

Частотно-модулированный сигнал с логарифмической разверткой по частоте (ЛогЧМ).

Частотно-модулированный сигнал с логарифмической разверткой по частоте (ЛогЧМ) представляет собой синусоиду с логарифмически увеличивающейся по времени частотой.

ЛогЧМ рассчитывается по формуле:

$$A = A_0 \sin \left( \frac{\omega T}{\ln \left( \frac{f_k}{f_0} \right)} \left( \frac{f_k}{f_0} \right)^{\frac{t}{T}} + \varphi_0 \right)$$

Где  $A_0 = \frac{2V_{\text{rms}}}{\sqrt{2}}$  - фаза сигнала,  $\varphi_0$  - начальная фаза сигнала,  $T$  - время изменения частоты,  $t$  - текущее время,  $f_0$  - начальная частота сигнала,  $f_k$  -



конечная частота сигнала.

## 9.7. Пилообразный сигнал

**Пилообразный сигнал** - компонент. Позволяет генерировать пилообразный сигнал определенного типа по заданным параметрам: амплитуде, частоте и смещению постоянной составляющей.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

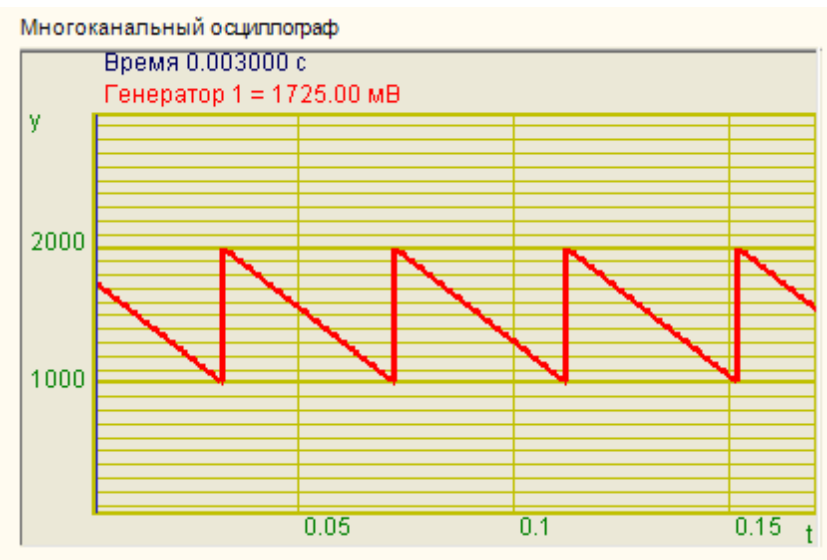
#### ➤ Входные:

- Частота - параметр, который задает значение частоты, Гц.
- Уровень - параметр, который задает амплитуду выходного сигнала, В.
- Смещение - параметр, позволяющий задавать смещение постоянной составляющей, В.
- Цикличность - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- Тип пиры - выбор типа пилообразного сигнала. На рисунках ниже представлены все 3 типа.

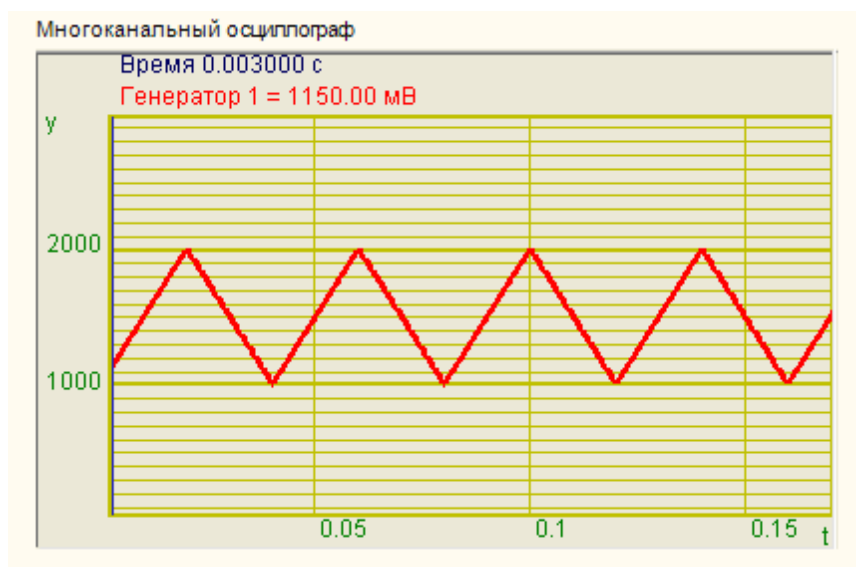
1. **Нарастающий тип** характеризуется интервалом нарастания



*Нарастающий тип*



*Ниспадающий тип*



- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

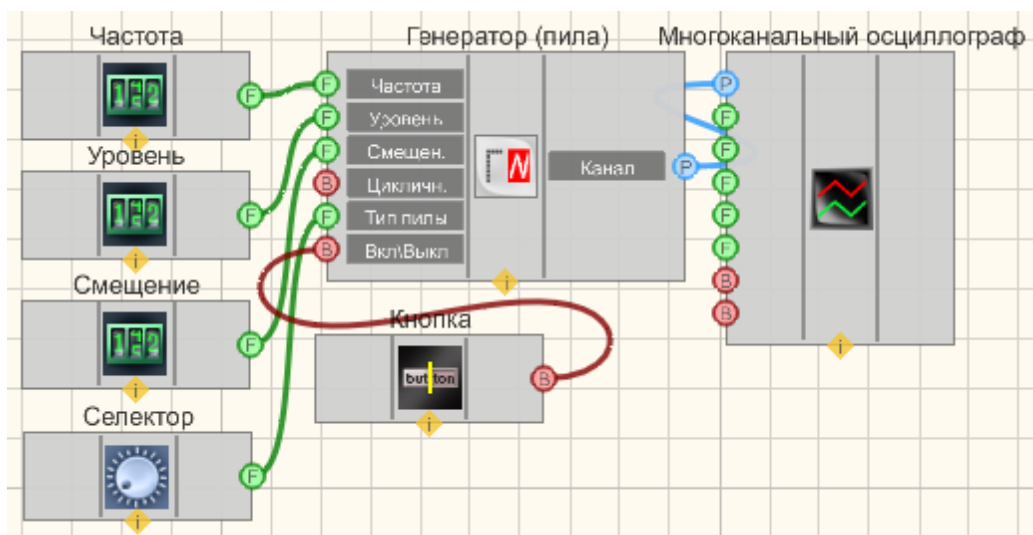
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Frequency (100) - частота выходного сигнала.
- Level (1) - амплитуда выходного сигнала, В.
- Shift (0) - смещение постоянной составляющей, В.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Cyclic (true) - параметр, отвечающий за число воспроизведений сигнала. Либо постоянное воспроизведение сигнала в цикле либо его однократное воспроизведение.
- pilature (нарастающий) - тип выводимого сигнала (нарастающий, ниспадающий, треугольный).
- Number (0) - номер генератора в системе, начиная с нулевого.



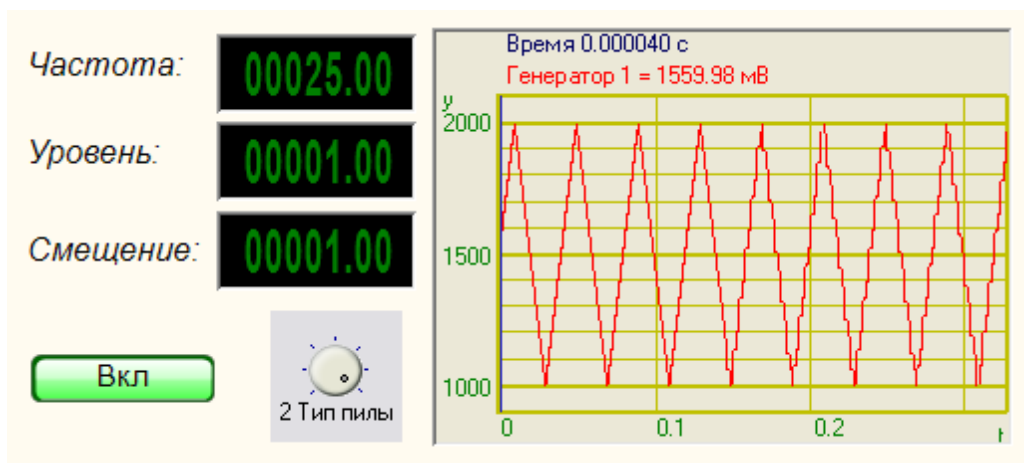
### Пример

### Проект в SCADA ZETView



В этой схеме компонент [Пилообразный сигнал](#)<sup>[572]</sup> создает пилообразный сигнал, с параметрами, которые задаются при помощи [Селекторов](#)<sup>[213]</sup>. Для просмотра сгенерированного сигнала служит компонент [Многоканальный осциллограф](#)<sup>[494]</sup>. Дополнительный [Селектор](#)<sup>[217]</sup> позволяет устанавливать тип пила (нарастающий, ниспадающий, треугольный). Результат работы составленной программы можно увидеть в интерфейсе оператора. [Кнопка с фиксацией](#) служит для включения или отключения.

### Результат работы проекта



Данный пример непосредственно в ZETView



### Математическое описание

Периодические сигналы – это сигналы, повторяющие сами себя через определенные промежутки времени  $T$ . Величина  $T$  называется периодом сигнала  $f(t)$ .

Любой периодический сигнал с периодом  $T$  можно представить в виде суммы гармонических сигналов. При этом самая низкая (основная) частота будет равна  $1/T$ . Все остальные частоты составляющих сигнала (гармоник) являются произведениями этой частоты на целые числа.

Если сигнал  $f(t)$  является периодическим с периодом  $T$ , то его можно представить в виде ряда Фурье:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(n\omega t) + b_n \sin(n\omega t)]$$

Где  $\omega$  называется основной (угловой) частотой сигнала:

$$\omega = \frac{2\pi}{T} = 2\pi f.$$

Коэффициенты Фурье  $a_n$  и  $b_n$  определяются следующими выражениями:

$$a_n = \frac{2}{T} \int_0^T f(t) \cos(n\omega t) dt$$

$$b_n = \frac{2}{T} \int_0^T f(t) \sin(n\omega t) dt$$

При  $n=0, 1, 2, \dots$

- Величина

$$\frac{a_0}{2} = \frac{1}{T} \int_0^T f(t) dt$$

является средним значением сигнала за один период, т.е. постоянной составляющей сигнала. Отметим, что  $b_0$  всегда равен нулю.

- Тригонометрическое представление ряда Фурье зависит от

значения сигнала в выбранный начальный момент времени  $t=0$ .

Кососимметричный треугольный сигнал, постоянная составляющая отсутствует:

$$f(t) = A \frac{2}{\pi} \sum_{k=1}^{\infty} \frac{\sin(2\pi kft)}{k} = A \frac{2}{\pi} \left[ \sin(\omega t) + \frac{1}{2} \sin(2\omega t) + \frac{1}{3} \sin(3\omega t) + \dots \right]$$

Подобные сигналы широко применяются в аппаратуре связи, телевидении, радиолокации. Наиболее часто их используют для создания временной развертки луча в электронно-лучевых трубках осциллографов, телевизоров и т. п. Другой важной областью применения пилообразного напряжения является преобразование напряжения во временной интервал в устройствах фазоимпульсной модуляции сигналов, при сравнении токов и напряжений и при замене напряжения цифровым кодом и т. п.

## 9.8. Радиоимпульсный сигнал

**Радиоимпульсный сигнал** - компонент. Позволяет генерировать радиоимпульсы (радиоимпульсом называют ограниченный во времени синусоидальный сигнал) с заданной частотой заполнения, уровнем, частотой следования и длительностью.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Частота заполнения - частота радиоимпульса внутри пакета, Гц.

- Уровень - параметр, который задает амплитуду выходного сигнала, В.
  - Частота следования - параметр, который задает частоту появления радиоимпульсов, Гц.
  - Длительность - длительность радиоимпульсного сигнала в периодах (число периодов).
  - Цикличность - параметр, отвечающий за число воспроизведений сигнала.
    1. True - постоянная генерация радиоимпульсов.
    2. False - генерация единичного импульса.
  - Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - вывод уже сгенерированного по заданным параметрам гармонического сигнала.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

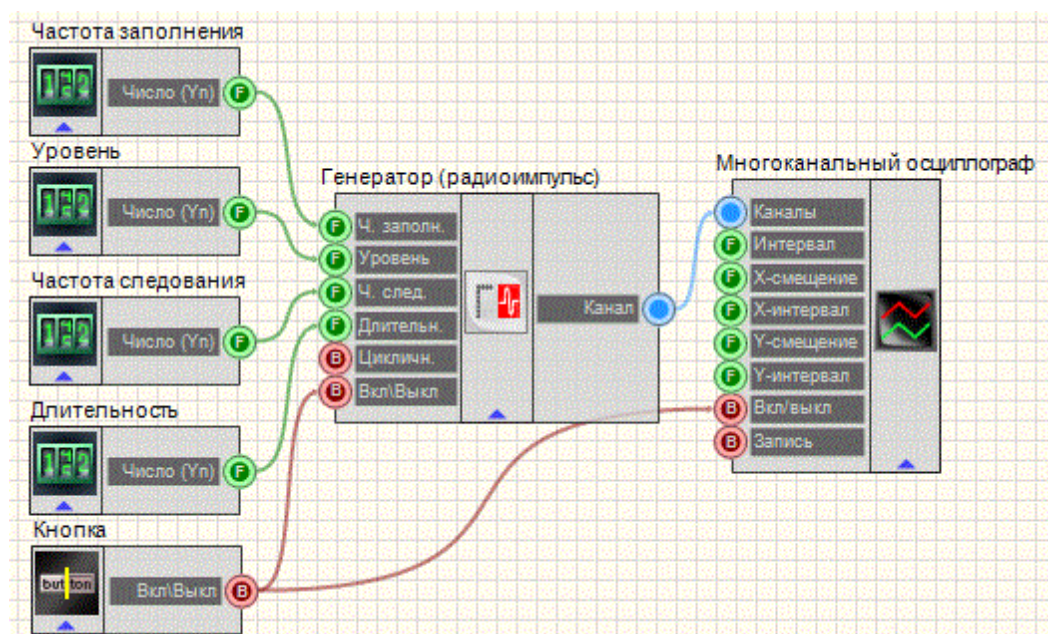
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FillFrequency (100) - частота заполнения сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- FollowFrequency (1) - частота следования сигнала, Гц.
- Duration (1) - длительность сигнала в периодах (число периодов).
- Cyclic (true) - число раз воспроизведения сигнала (режим постоянной генерации сигналов/единственный импульс).
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.



### *Пример*

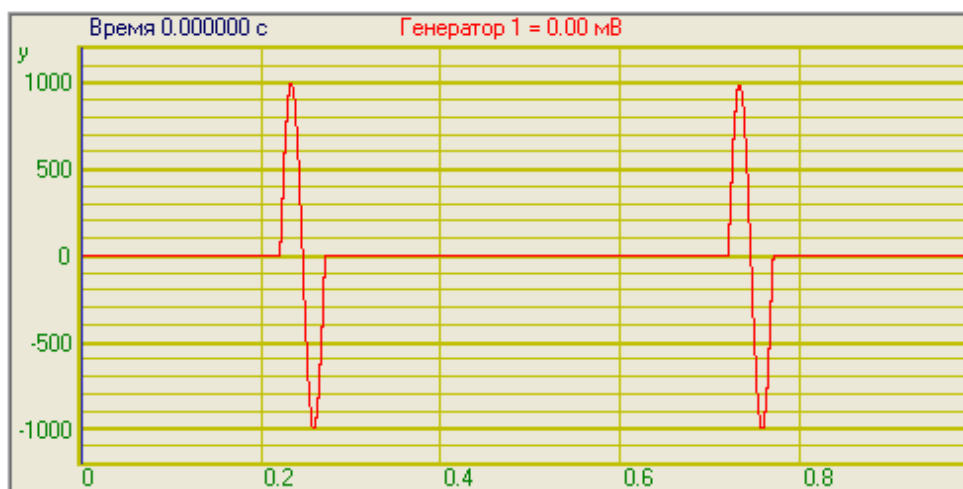
### *Проект в SCADA ZETView*



В этой схеме компонент [Радиоимпульсный сигнал](#)<sup>[244]</sup> создает радиоимпульсный сигнал, с параметрами, которые задаются при помощи [Селекторов](#)<sup>[213]</sup>. Для просмотра сгенерированного сигнала служит компонент [Многоканальный осциллограф](#)<sup>[494]</sup>. Результат работы составленной программы можно увидеть в интерфейсе оператора. [Кнопка с фиксацией](#) служит для включения или отключения.

### *Результат работы проекта*





Частота следования:

00002.00

Уровень:

00001.00

Частота заполнения:

00020.00

Длительность:

00001.00

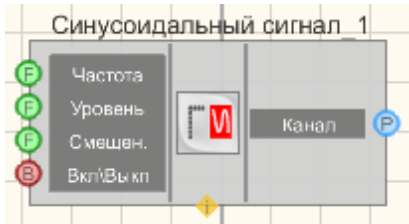
## 9.9. Синусоидальный сигнал

**Синусоидальный сигнал** - компонент. Позволяет генерировать гармонический сигнал по заданным параметрам: амплитуде, частоте и смещению постоянной составляющей. Для работы компонента необходимо наличие физического ZET-устройства, содержащего ЦАП, подключенного к ПК.



*Примечание: Можно запустить столько программ в SCADA или в ZETLab, сколько подключено генераторов ЦАП, в противном случае программы будут работать неправильно.*

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Частота - параметр, который задает значение частоты, Гц
- Уровень - параметр, который задает амплитуду выходного сигнала, В.
- Смещение - параметр, позволяющий задавать смещение постоянной составляющей, В.
- Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.

#### ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам гармонического сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Frequency (100) - частота выходного сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Shift (0) - смещение постоянной составляющей, В.
- Activate (false)- состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.
- sinusnumber (1) - номер выбранного синусоидального канала (при создании сигнала сложной формы (при помощи [Суперпозиции генераторов](#)<sup>[251]</sup>) может потребоваться более одного генератора синусоидального сигнала, и чтобы их можно было различить, устанавливается данный параметр).



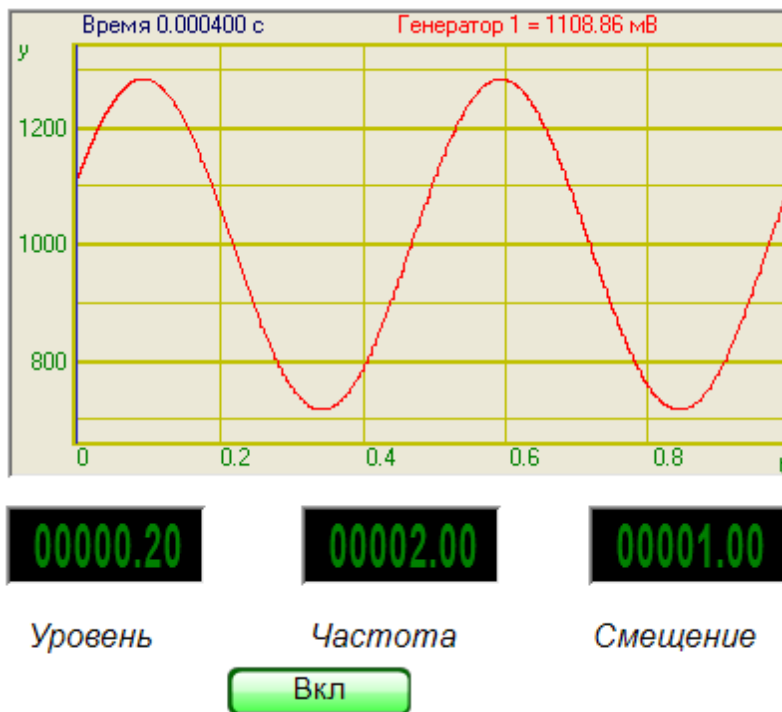
### Пример

### Проект в SCADA ZETView



В этой схеме компонент [Синусоидальный сигнал](#)<sup>[247]</sup> создает синусоидальный сигнал, с параметрами, которые задаются при помощи [Селекторов](#)<sup>[215]</sup>. [Гумблеры](#)<sup>[368]</sup> позволяют включать и выключать [Многоканальный осциллограф](#)<sup>[494]</sup> и [Синусоидальный сигнал](#)<sup>[247]</sup>. Для просмотра сгенерированного сигнала служит компонент [Многоканальный осциллограф](#)<sup>[494]</sup>. Результат работы составленной программы можно увидеть в интерфейсе оператора. [Кнопка с фиксацией](#) служит для включения или отключения.

### Результат работы проекта



Данный пример непосредственно в ZETView



### Математическое описание

Для описания цепей переменного тока чаще всего используется функции синуса и косинуса. Функция синуса имеет следующий вид:

$$A(t) = A_m \sin(\omega t + \varphi_0)$$

Величина  $A(t)$  называется мгновенным или фактическим значением функции,  $\omega$  - угловой частотой,  $\varphi_0$  - фазовым сдвигом. Синус является периодической функцией с периодом, равным  $2\pi$ .

Периодом  $T$  называется интервал времени между двумя идентичными значениями функции. Частота  $f$  функции синуса является величиной, обратной периоду:

$$T = \frac{2\pi}{\omega}; f = \frac{1}{T}; \omega = 2\pi f$$

Функция косинуса аналогична функции синуса. Ее можно записать как:

$$A = A_m \cos\varphi$$

Функции синуса и косинуса связаны между собой следующим соотношениями:

$$\sin\varphi = \cos\left(\frac{\pi}{2} - \varphi\right)$$

$$\cos\varphi = \sin\left(\frac{\pi}{2} + \varphi\right)$$

## 9.10. Синхронный генератор

**Синхронный генератор** - компонент. Позволяет генерировать сигнал с любым количеством каналов (до 20) с заданной частотой периодического сигнала. Для работы компонента необходимо наличие физического ZET-устройства, содержащего ЦАП, подключенного к ПК.



**Примечание:** Можно запустить столько программ в SCADA или в ZETLab, сколько подключено генераторов ЦАП, в противном случае программы будут работать неправильно.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Частота - параметр, который задает значение частоты периодического сигнала, Гц
- Вкл\Выкл - параметр, отвечающий за управление состоянием работы синхронного генератора.

## ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам гармонического сигнала.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

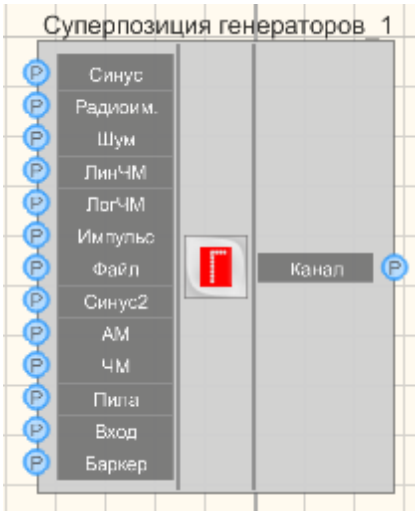
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Quantity (1) - количество каналов синхронного генератора.
- Frequency (1) - частота периодического сигнала синхронного генератора, Гц.

## 9.11. Суперпозиция генераторов

**Суперпозиция генераторов** - компонент. Позволяет создавать результирующий сигнал, из набора входных (пилообразного, импульсного и других). Служит для создания новых, требуемой формы сигналов.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Синус - служит для подключения синусоидального сигнала.
- Радиои́мпульс - служит для подключения радиоимпульсного сигнала.
- Шум - служит для подключения шумового сигнала.
- ЛинЧМ - служит для подключения линейного частотно-модулированного сигнала.
- ЛогЧМ - служит для подключения логарифмически частотно-модулированного сигнала.
- Импульс - служит для подключения сигнала, состоящего из прямоугольных импульсов.
- Файл - служит для подключения сигнала, воспроизводимого из файла.
- Синус2 - служит для подключения дополнительного синусоидального сигнала.
- АМ - служит для подключения амплитудно-модулированного сигнала.
- АМ - служит для подключения частотно-модулированного сигнала.
- Пила - служит для подключения пилообразного сигнала.
- Вход - служит для подключения Виртуального канала или Измерительного канала.
- Баркер - служит для подключения генератора сигнала по кодам Баркера.

#### ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

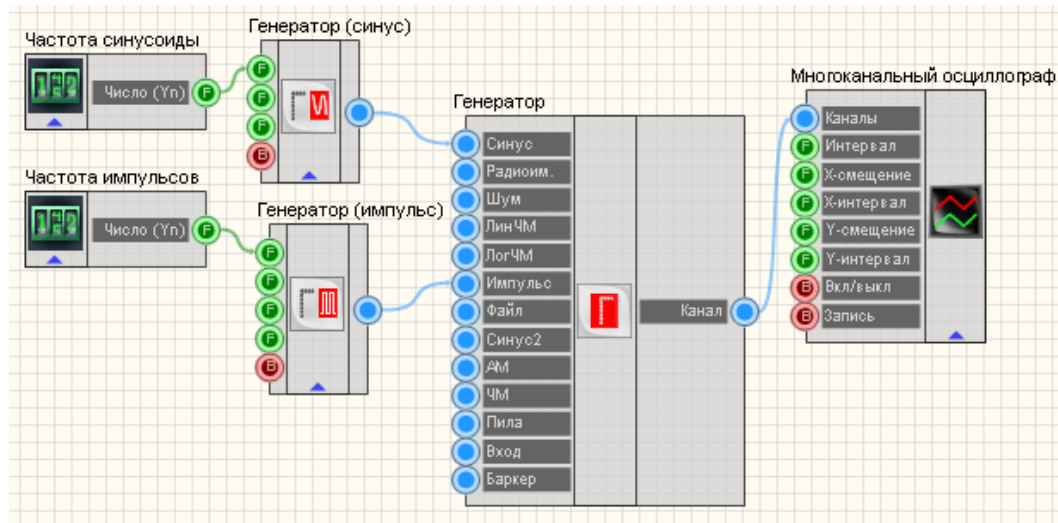
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Number (0) - номер генератора в системе, начиная с нулевого.



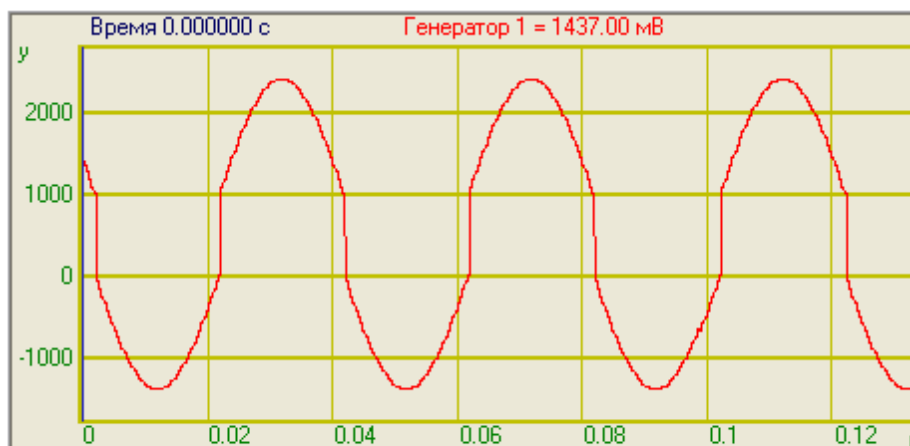
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Суперпозиция генераторов](#)<sup>[251]</sup> создает результирующий сигнал, состоящий из суммы синусоидального и импульсного сигналов. Для создания данных сигналов используются компоненты [Синусоидальный сигнал](#)<sup>[247]</sup> и [Импульсный сигнал](#)<sup>[244]</sup>. [Селекторы](#)<sup>[213]</sup> нужны для установки значений частоты. [Многоканальный осциллограф](#)<sup>[494]</sup> служит для графического отображения полученного сигнала.

### Результат работы проекта



Частота синусоиды:

00025.00

Частота импульсов:

00025.00

Данный пример непосредственно в ZETView



### Математическое описание

В 50 – 60-е годы XX столетия были разработаны целые классы дискретных сигналов с совершенными корреляционными свойствами.

Автокорреляционная функция фазомодулированных сигналов имеет вид типичный для всех типов широкополосного сигнала. Нормированная автокорреляционная функция состоит из центрального (основного) типа с амплитудой 1, размещенного на интервале  $(-\tau, \tau)$  и боковых (фоновых) максимумов, распределенных на интервале  $(-T, \tau)$  и  $(\tau, T)$ .

Амплитуды боковых типов принимают различные значения, но у сигналов с “хорошей” корреляцией они малы, т.е. существенно меньше амплитуды центрального пика. Отношение амплитуды центрального пика (в данном случае 1) к максимальной амплитуде боковых максимумов называют коэффициентом подавления  $K$ . Для произвольных широкополосных сигналов



с базой В

$$K \approx \frac{1}{\sqrt{B}}$$

Для фазомодулированных широкополосных сигналов  $K \approx \frac{1}{\sqrt{N}}$ . Пример автокорреляционной функции широкополосного сигнала дан на рисунке 2. Величина К существенно зависит от вида кодовой последовательности А. При правильном выборе закона формирования А можно добиться максимального подавления, а в ряде случаев – равенства амплитуд всех боковых максимумов.

Сигналы Баркера имеют спектры мощности, наименее уклоняющиеся (в квадратичном смысле) от спектра единичного дискрета. Их автокорреляционные функции соответственно приближаются к автокорреляционной функции дискрета. Ниже приводятся таблица кодов Баркера.

N	k												Уровень боковых лепестков	
	1	2	3	4	5	6	7	8	9	10	11	12		
2	+1	-1												-
3	+1	+1	-1											-1/3
4	+1	+1	-1											1/4
5	+1	+1	+1	+1										1/5
7	+1	+1	+1	-1	+1	-1								-1/7
11	+1	+1	+1	-1	-1	+1	-1	-1	+1	-1				-1/11
13	+1	+1	+1	+1	-1	-1	+1	+1	-1	+1	-1	+1		1/13

Табл.1

Кодовая последовательность сигнала Баркера состоит из символов  $\pm 1$  и характеризуется нормированной автокорреляционной функцией вида:

$$B(\tau) = \begin{cases} 1 & \text{для } \tau = 0, \\ 0 & \text{для } \tau = 2l + 1, \\ \pm \frac{1}{N} & \text{для } \tau = 2l. \end{cases} \quad (1)$$

$$\text{Где } l = 0, 1, \dots, \frac{N-1}{2}$$

Знак в последней строчке зависит от величины  $N$ . На рисунках 1-2 показаны фазомодулированный сигнал, его комплексная огибающая и автокорреляционная функция семизначного кода Баркера.

Из (1) следует, что одна из особенностей сигнала Баркера - равенство амплитуд всех  $(N-1)$  боковых максимумов АКФ, и все они имеют минимально возможный уровень, не превышающий  $1/N$ . В таблице 1 приведены известные кодовые последовательности Баркера и их уровни боковых типов АКФ. Кодовые последовательности, обладающие свойствами  $V(\tau)$ , для  $N > 13$  не найдены.

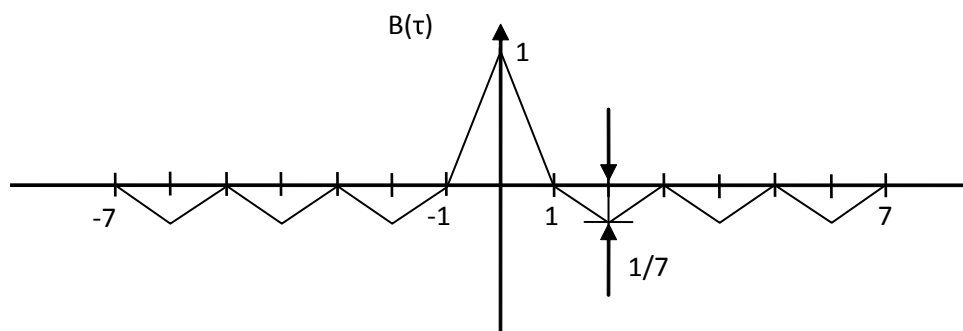


Рис.1 – Автокорреляционная функция семизначного кода Баркера.

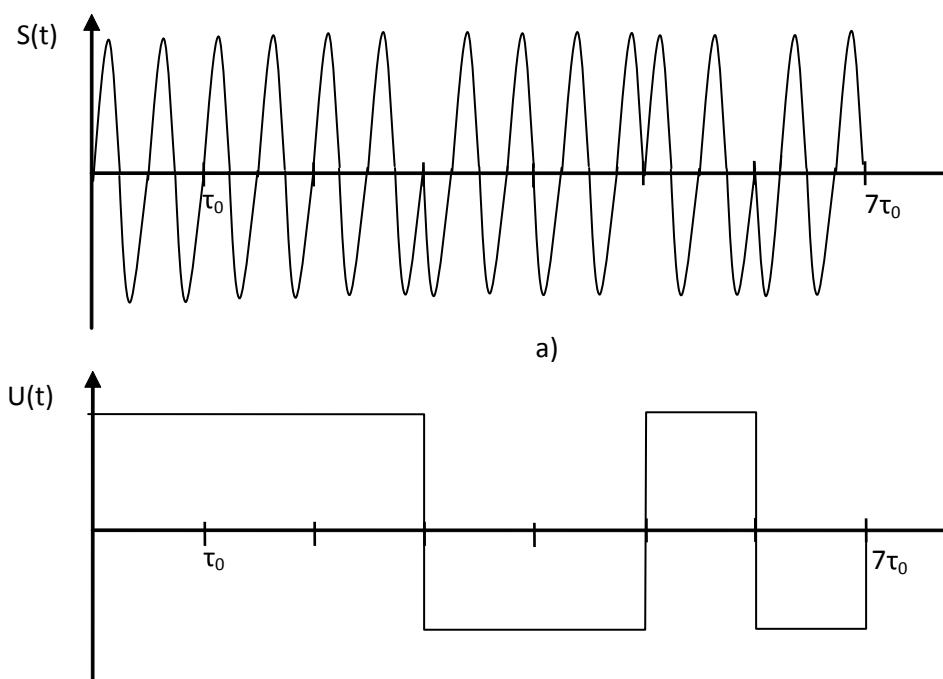


Рис.2 –а) Фазомодулированный сигнал, б) его комплексная огибающая.

Коды Баркера обладают наилучшими среди известных псевдослучайных последовательностей свойствами шумоподобности, что и обусловило их широкое применение: именно они используются в протоколе беспроводных сетей IEEE 802.11. В протоколах семейства 802.11 используется код Баркера длиной в 11 чипов (11100010010). Для того чтобы передать сигнал логическая единица передается прямой последовательностью Баркера, а логический ноль – инверсной последовательностью.

## 9.12. Частотная модуляция

**Частотная модуляция** - компонент. Позволяет генерировать сигнал, создаваемый по специальному закону модуляции частоты, при котором информационный сигнал управляет частотой несущего колебания. По сравнению с амплитудной модуляцией здесь амплитуда остаётся постоянной.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входящие:

- Несущая частота - частота несущего сигнала, Гц.
- Модуляционная частота - частота частотной модуляции, Гц.
- Уровень - параметр, который задает амплитуду выходного сигнала, В.
- Глубина - параметр, задающий глубину частотной модуляции.
- Вкл/Выкл - параметр, отвечающий за включение и выключение генератора.

#### ➤ Выходные

- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

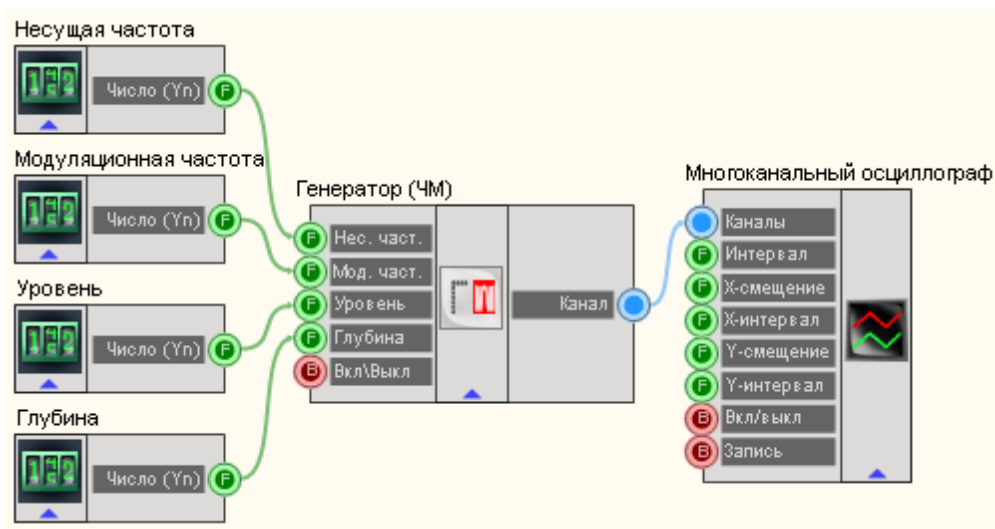
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- CentralFrequency (1000) - частота несущего сигнала, Гц.
- ModFrequency (1) - частота частотной модуляции, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- Depth (0) - глубина частотной модуляции.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.



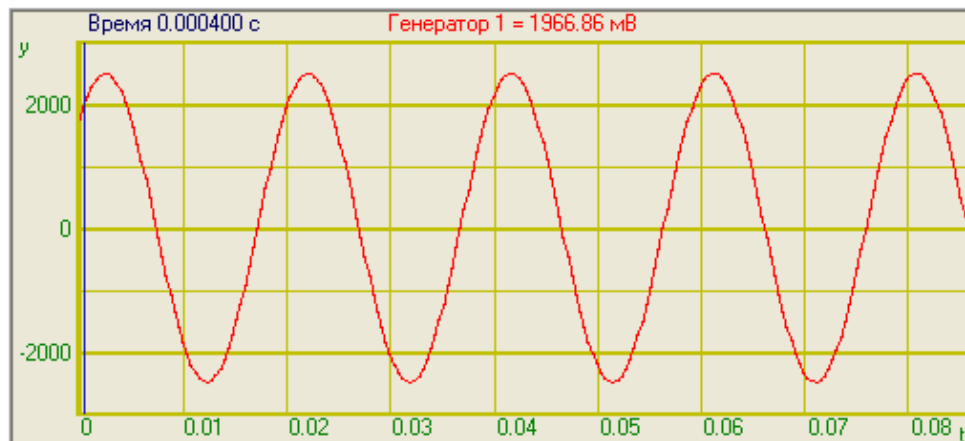
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Частотная модуляция](#)<sup>[257]</sup> на несущей производит частотное модулирование сигнала. [Селекторы](#)<sup>[213]</sup> служат для задания несущей и модуляционной частот, уровня, глубины. [Многоканальный осциллограф](#)<sup>[494]</sup> нужен для визуального представления смодулированного сигнала.

### Результат работы проекта



Несущая частота:

00082.00

Уровень:

00001.00

Модуляционная частота:

00026.00

Глубина:

00000.50

Данный пример непосредственно в ZETView



### *Математическое описание*

Фазовая и Частотная модуляция сигнала заключаются в изменении частоты или фазы ВЧ колебания по закону управляющего сигнала. ЧМ была известна в 20х годах, но практическое применение получила с середины 30х годов, с развитием техники УКВ связи. В настоящее время ЧМ широко используется в коммерческой радиосвязи, радиовещании на метровых волнах и телевидении для передачи звукового сопровождения и т.д. Как известно, периодические колебания можно выразить уравнением:

$$i = J_m \cos (\omega_0 t + \varphi),$$

Где  $\varphi$  - фаза колебания, характеризующая состояние колебательного процесса в данный момент времени,  $\omega_0$  - частота ВЧ колебаний. Фаза является функцией времени и связана с частотой зависимостью:

$$\varphi = \int \omega dt$$

В свою очередь:

$$\omega = \frac{d\varphi}{dt}$$

Взаимная зависимость фазы и угловой частоты такова, что всякое отклонение фазы от линейного закона приводит к отклонению частоты от начального значения и наоборот. По этим причинам при ФМ всегда меняется частота, а при Частотной модуляции – фаза.

Несмотря на их тесную связь, частотную и фазовую модуляции можно различать по тому, какой из параметров (частота или фаза) находится под воздействием модулирующего фактора. Частотная модуляция намного лучше, чем амплитудная модуляция.

Если модулирующий фактор меняет фазу, то модуляцию следует считать фазовой (хотя при этом меняется и частота) и, наоборот, если модулирующий фактор меняет частоту (хотя при этом меняется фаза), то модуляцию следует считать частотной.

Рассмотрим их взаимную зависимость. При ФМ фаза меняется по закону модулирующего напряжения  $U\Omega(t)$ , т.е.  $\varphi = \omega_0 t + \varphi_0$ ,

Где  $\omega_0 t$  - составляющая фазы, изменяющаяся по линейному закону;

$\varphi_0 = kU_{m\Omega} \cos \Omega t$  - составляющая фазы, изменяющаяся по закону напряжения  $U_{\Omega}(t)$ , где  $k$  – коэффициент пропорциональности.

В результате при модуляции по фазе:

$$i = J_m \cos(\omega_0 t + kU_{m\Omega} \cos \Omega t) \text{ или } i = J_m \cos(\omega_0 t + m_\varphi \cos \Omega t),$$

где  $m_\varphi = kU_{m\Omega} = \Delta\varphi$  - максимальное отклонение фазы от линейного закона или индекс фазовой модуляции. Индекс ФМ, как и коэффициент модуляции (при амплитудной модуляции) зависит от амплитуды низкой частоты  $U_{m\Omega}$ :

Изменение фазы приводит к изменению частоты: при опережении фазы частота  $\omega$  увеличивается, а при отставании фазы - уменьшается.

Закон изменения частоты при ФМ можно определить из уравнения

$$\omega = \frac{d\varphi}{dt} = \frac{d(\omega_0 t + m_\varphi \cos \Omega t)}{dt} = \omega_0 - m_\varphi \Omega \sin \Omega t = \omega_0 - \Delta\omega \sin \Omega t,$$

Где  $\Delta\omega = m_\varphi \Omega = kU_{m\Omega} \Omega$  - максимальное отклонение частоты от начального значения или девиация частоты. Из этого выражения следует, что при ФМ прямой пропорциональности между девиацией частоты  $\Delta\omega$  и амплитудой модулирующего напряжения  $U_{m\Omega}$  не соблюдается. Девиация частоты  $\Delta\omega$

зависит от частоты  $\Omega$  модуляции и различна на различных модулирующих частотах. Эта зависимость объясняется так: чем больше частота модуляции  $\Omega$ , тем чаще (быстрее) меняется фаза  $\varphi$  и, следовательно, больше скорость изменения фазы во времени (т.е. угловая частота выше). При ЧМ частота колебаний меняется по закону модулирующего напряжения:  $\omega = \omega_0 + \omega_1$ ,

где  $\omega_1 = kU_{m\Omega} \cos \Omega t$  – характеризует отклонение частоты от начального значения. Величина  $\Delta\omega = kU_{m\Omega} \Omega$  - есть максимальное отклонение частоты от среднего значения или девиация частоты. В моменты максимума, т.е.  $+U_{m\Omega}$ , частота увеличивается, а в моменты минимума -  $-U_{m\Omega}$  частота уменьшается.

При этом меняется фаза колебаний. Определим фазу колебаний для частотной модуляции:

$$\varphi = \int \omega dt = \int (\omega_0 + \Delta\omega \cos \Omega t) dt = \omega_0 t + \frac{\Delta\omega}{\Omega} \sin \Omega t + \varphi_0.$$

Полагая для упрощения  $\varphi_0 = 0$ , получим:

$$\varphi = \omega_0 t + \frac{\Delta\omega}{\Omega} \sin \Omega t = \omega_0 t + m_f \sin \Omega t$$

Где  $m_f = \frac{\Delta\omega}{\Omega} = \Delta\varphi$  - максимальное отклонение фазы от линейного закона или индекс частотной модуляции. Следовательно, как видим, из выше приведенной формулы для  $\varphi$  – фаза колебаний при частотной модуляции меняется по иному закону, чем частота и не пропорциональна амплитуде модулирующего напряжения  $U_m \Omega$ . На основании уравнения при ЧМ колебания можно представить следующим образом:

$$i = J_m \cos(\omega_0 t + m_f \sin \Omega t)$$

Форма ЧМ сигнала внешне похожа на ФМ колебания. Однако, законы изменения частоты в них различны. Так, если фаза меняется по закону  $\sin \Omega t$ , то частота – по закону  $\cos \Omega t$  и наоборот.

### 9.13.Шум

**Шум** - компонент. Позволяет генерировать стационарный шум, спектральные составляющие которого равномерно распределены по всему диапазону задействованных частот.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Начальная частота - параметр, который служит для задания начальной частоты частотного диапазона, в котором будет генерироваться белый шум, Гц.
- Конечная частот - параметр, который служит для задания конечной частоты



частотного диапазона, в котором будет генерироваться белый шум, Гц.

- Уровень - параметр, который задает амплитуду выходного сигнала, В.
  - Тип - параметр, позволяющий задавать вид шума:
    1. Белый.
    2. Полосовой.
    3. Розовый.
    4. Детерминированный.
  - Вкл\Выкл - параметр, отвечающий за включение и выключение генератора.
- Выходные
- Канал - вывод уже сгенерированного по заданным параметрам сигнала.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

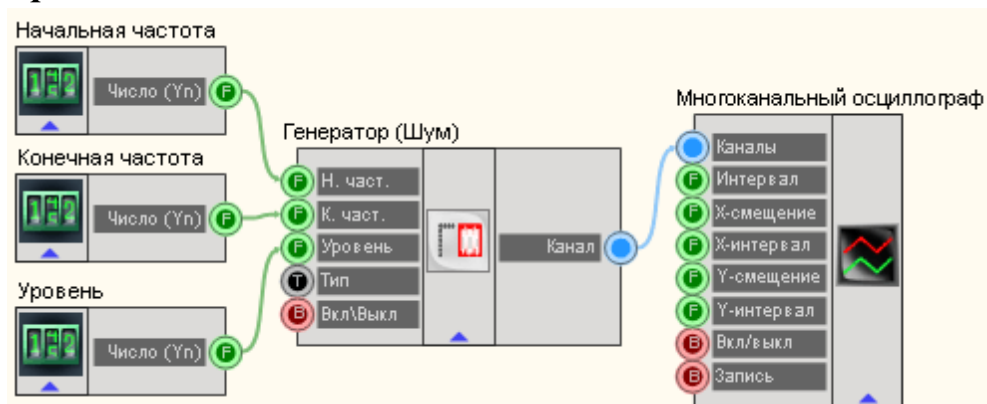
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- StartFrequency (0) - начальная частота выходного сигнала, Гц.
- EndFrequency (1000) - конечная частота выходного сигнала, Гц.
- Level (1) - амплитуда выходного сигнала, В.
- type (Белый) - параметр, позволяющий задавать вид шума:
  1. Белый.
  2. Полосовой.
  3. Розовый.
  4. Детерминированный.
- Activate (false) - состояние подачи сигнала (подан или отключен).
- Number (0) - номер генератора в системе, начиная с нулевого.



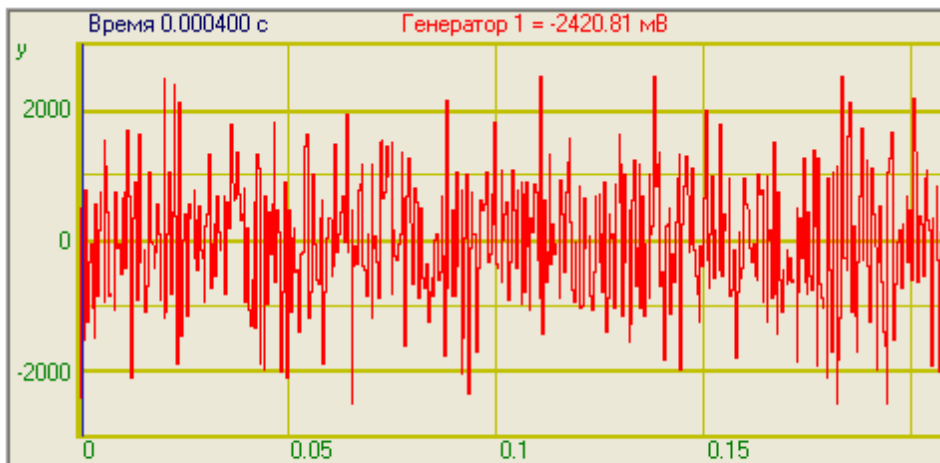
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Шум](#)<sup>[262]</sup> генерирует белый шум. [Селекторы](#)<sup>[213]</sup> служат для задания начальной, конечной частоты, уровня шума.

### Результат работы проекта



Начальная частота:

00019.00

Конечная частота:

00094.00

Уровень:

00001.00

Данный пример непосредственно в ZETView



### Математическое описание

Белый шум – стационарный шум, спектральные составляющие которого равномерно распределены по всему диапазону задействованных частот.

Белый шум рассчитывается по формуле:

$$A = A_0 \left( \sum_1^{12} \text{rand}() - 6 \right),$$

Где  $A_0 = \frac{2V_{\text{rms}}}{\sqrt{2}}$  – начальная амплитуда сигнала.

Rand() – функция вычисления случайного числа, причем  $0 < \text{rand}() < 1$ .

Термин “белый шум” обычно применяется к сигналу, имеющему автокорреляционную функцию\*, математически описываемую дельта-функцией Дирака\*\* по всем измерениям многомерного пространства, в котором этот сигнал рассматривается. Сигналы, обладающие этим

свойством, могут рассматриваться как белый шум. Данное статистическое свойство является основным для сигналов такого типа.

Полосовой шум - шумовой сигнал с ограниченным частотным интервалом.

Розовый шум - шумовой сигнал, спектральный уровень которого снижается с увеличением частоты со спадом 3 дБ на октаву.

Детерминированный шум - белый шум ограниченный заданным частотным диапазоном.

-----  
 -----

\* В обработке сигналов автокорреляционная функция определяется интегралом

$$\psi(\tau) = \int f(t)f(t - \tau)dt$$

и показывает связь сигнала (функции  $f(t)$ ) с копией самого себя, смещенного на величину  $\tau$ .

\*\*  $\delta$ -функция Дирака с областью определения  $R^n$  для точки  $a$  определяется формальным соотношением

$$(\delta; f) = \int_{R^n} \delta(\bar{x} - \bar{a})f(\bar{x}) d^m x$$

\*\*\*

Почти в любой области измерения значение предельно различимого слабого сигнала определяется шумом – мешающим сигналом, который забивает полезный сигнал. От некоторых шумов невозможно избавиться (флуктуации измеряемой величины). От некоторых можно избавиться при помощи фильтров. Термин “шум” применяется ко всему тому, что маскирует полезный сигнал. “Шумом” может оказаться другой электрический сигнал (“помеха”), но чаще это случайный шум физической природы.

*Джонсоновский шум.* Любой резистор на плате генерирует на своих выводах некоторое напряжение шума, известное как “шум Джонсона” (Тепловой шум). У него горизонтальный частотный спектр, т.е. одинаковая мощность шума на всех частотах. Шум с горизонтальным спектром называют “белым шумом”. Реальное напряжение шума в незамкнутой цепи, порожденное сопротивлением  $R$ , находящимся при температуре  $T$ ,

выражается формулой:

$$U_{ш.эф.} = U_{ш.Р} = \sqrt{4kTRB}$$

Где  $k$  - постоянная Больцмана,  $T$  – абсолютная температура в Кельвинах,  $B$  – полоса частот в Гц.

Амплитуда напряжения шума Джонсона, вообще говоря, в данный момент непредсказуема, но она подчиняется распределению Гаусса. Данный вид шума устанавливает нижнюю границу напряжения шумов любого детектора, источника сигнала, содержащего резисторы в своей схеме.

**Дробовый шум** – шум, возникающий в электрическом токе, вследствие движения дискретных носителей заряда. Конечность (квантованность) заряда приводит к статистическим флуктуациям тока. Если заряды действуют независимо друг от друга, то флуктуирующий ток определяется формулой:

$$I_{ш.эф.} = I_{ш.Р} = \sqrt{2qI_0B}$$

Где  $q$  – заряд электрона,  $I_0$  - постоянная составляющая (“установившееся значение”) тока,  $B$  – ширина полосы частот измерения.

Шум  $1/f$  (**фликкер-шум**). Дробовый и тепловые шумы – это неуменьшаемые виды шума, возникающие в соответствии с законами физики. Изготовленные по-разному резисторы имеют одинаковое значение теплового шума. Реальные резисторы подвержены флуктуациям сопротивления, которые порождают дополнительное напряжения шума, пропорциональное протекающему через резистор постоянному току. Этот шум имеет спектр, примерно описываемый зависимостью  $1/f$  (постоянная мощность на декаду частоты) и иногда называется “розовым шумом”. Примерами таких шумов является шум тока базы у транзистора и шум катодного тока в электровакуумных лампах. Данный вид шума также встречается в потоке песка в песочных часах, пассажирских потоках на скоростных железных дорогах, скорости океанических течений.

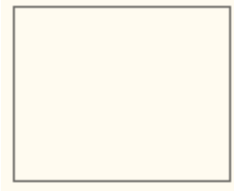
## Глава 10. Группы

Enter topic text here.

## 10.1.Объединить в группу

**Объединить в группу** - компонент. Предназначен объединения компонентов в логические блоки в режиме оператора.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
Не имеет.	

### Параметры:

➤ Входные:

Не имеет.

➤ Выходные:

Не имеет.

### Настраиваемые свойства:

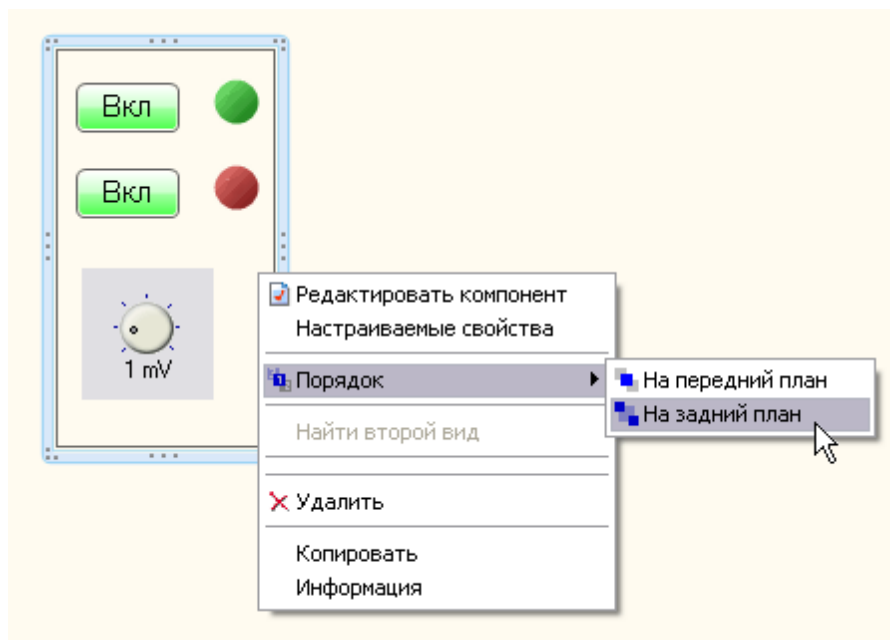
➤ Общие свойства (окружение):

Не имеет.

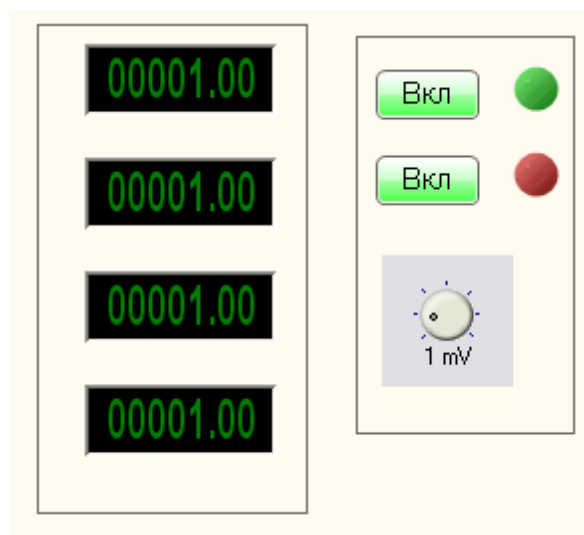
➤ Частные свойства:

Не имеет.

Иными словами этот компонент представляет собой просто рамку. Важно не забывать помещать его на задний план, либо компоненты, которые будут располагаться внутри рамки, помещать на передний план. Иначе рамка может закрывать другие компоненты, так как компоненты накладываются друг на друга в том порядке, в каком они помещены в рабочую область.



### Пример




## 10.2. Панель Windows стиль

**Панель Windows стиль** - компонент. Предназначен объединения компонентов в логические блоки в режиме оператора.

Или, иными словами "блок".

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
Не имеет.	

**Параметры:**

## ➤ Входные:

Не имеет.

## ➤ Выходные:

Не имеет.

**Настраиваемые свойства:**

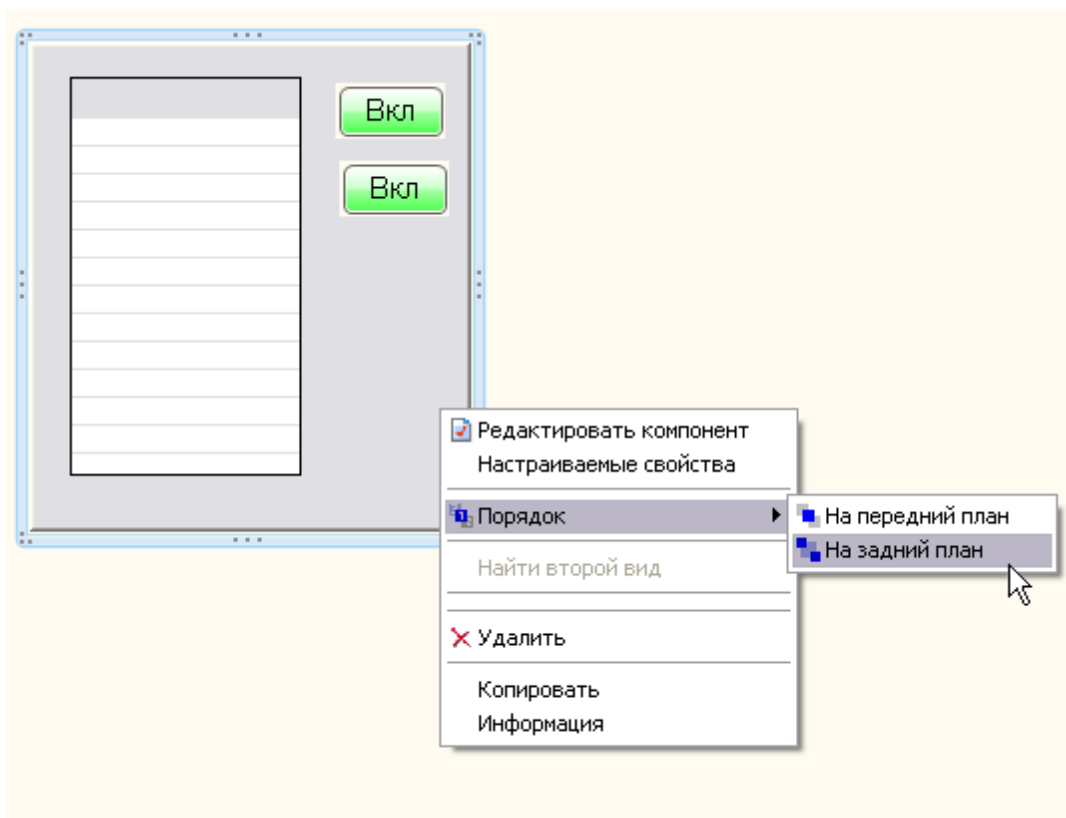
## ➤ Общие свойства (окружение):

Не имеет.

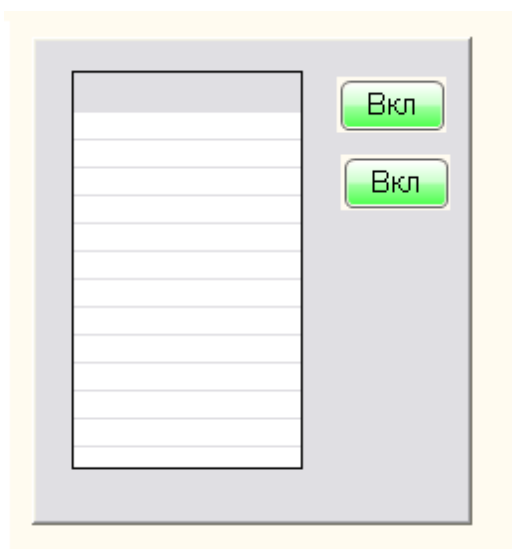
## ➤ Частные свойства:

Не имеет.

Иными словами этот компонент представляет собой просто блок. Важно не забывать помещать его на задний план, либо компоненты, которые будут располагаться внутри блока, помещать на передний план. Иначе блок может закрывать другие компоненты, так как компоненты накладываются друг на друга в том порядке, в каком они помещены в рабочую область.



### *Пример*



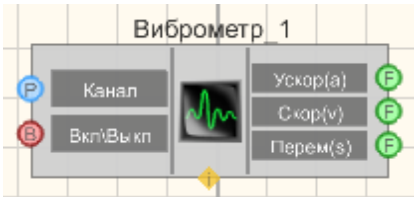


## Глава 11.Измерение (ВП)

### 11.1.Виброметр

**Виброметр** - компонент. Предназначен для измерения среднеквадратических и пиковых значений виброускорения, виброскорости и виброперемещения при помощи пьезодатчиков (акселерометров), подключенных к входным каналам анализаторов сигнала.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤Входные:

- Канал - канал, к которому подключается термосопротивление.
- Вкл\Выкл - параметр, отвечающий за включение и выключение термометра термопар.

##### ➤Выходные:

- Ускорение(a) - вывод, с которого поступает усредненное за установленное время значение ускорения.
- Скорость(v) - вывод, с которого поступает усредненное за установленное время значение скорости.
- Перемещение(s) - вывод, с которого поступает усредненное за установленное время значение перемещения.

#### Настраиваемые свойства:

##### ➤Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

##### ➤Частные свойства (в скобках значение, установленное по умолчанию):

- averagetime (0.1) - время, за которое подсчитывается усредненное значение, с:
  1. 0.1 сек.
  2. 1 сек.
  3. 10 сек.
- measuretype (Пик) - тип проводимых измерений, В:
  1. СКЗ - среднеквадратичное значение.

2. Пик - максимальное значение на протяжении всего времени работы (значение пика  $\geq$  значения амплитуды).

3. Амплитуда - максимальное значение смещения или изменения переменной величины от среднего значения.

- ReactionTime (0) - минимальное время сервера, после которого компонент начнет работать, с.
- Activate (true) - состояние работы (включено/выключено).
- fir (2 - 200) - фильтр пропускания полезного сигнала:
  1. 2 - 200 Гц.
  2. 10 - 1000 Гц.
  3. 3 - 10000 Гц.
  4. 2 - 10 Гц.

---

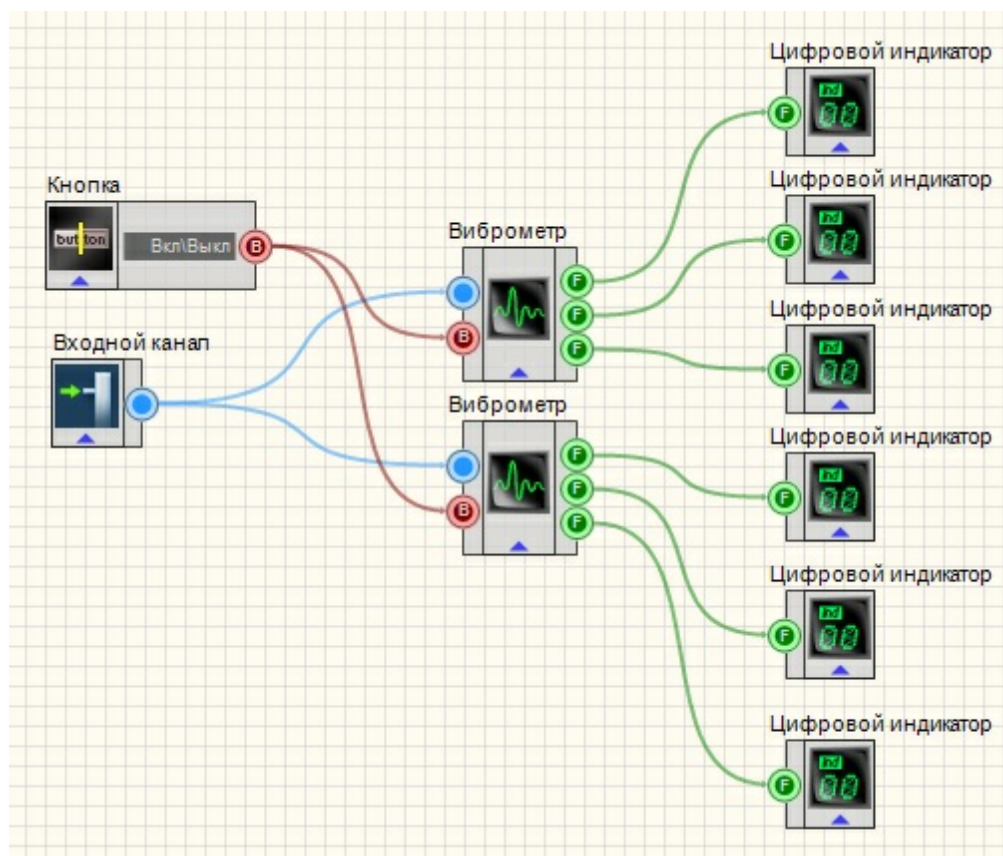
### ***Внимание!***

Некоторые измерительные программы (вольтметр постоянного тока, виброметр, а также, в будущем, некоторые другие) при загрузке с помощью интерфейса UNIT (например, в ZETView) могут работать в многоканальном режиме. В этом режиме программа может выполнять свои измерения не по одному, а одновременно по нескольким каналам (количеством не более 128). При этом у каждого канала программы могут быть свои параметры, такие как время усреднения, частота дискретизации канала, тип используемого полосового фильтра для виброметра и т.д. Результаты измерений доступны родительской программе через UNIT.



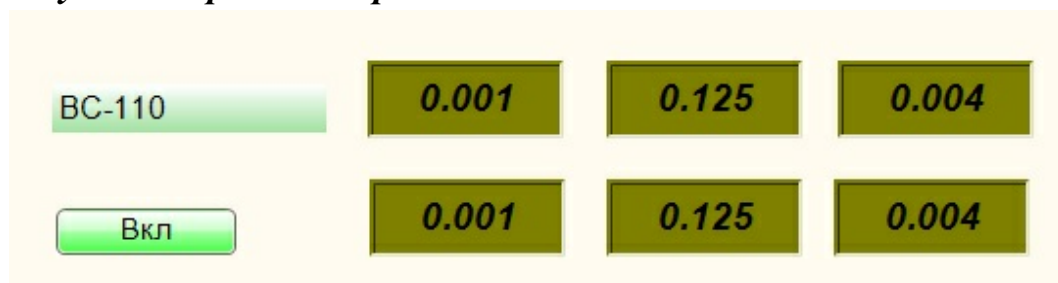
### ***Пример***

#### ***Проект в SCADA ZETView***



В приведенном проекте многоканальный компонент [Виброметр](#)<sup>[271]</sup> измеряет значения сигнала, поступающего с [Измерительного канала](#)<sup>[185]</sup>. [Кнопка с фиксацией](#)<sup>[366]</sup> служит для включения или отключения виброметров. [ЖК индикаторы](#)<sup>[348]</sup> служат для графического представления значения.

### Результат работы проекта



Данный пример непосредственно в ZETView

Более подробно о программе "Виброметр":

Виброметр

Программа предназначена для измерения параметров вибрации с вибродатчиков, подключенных к входным каналам анализаторов спектра.

В программе измеряются следующие параметры: виброускорение, виброскорость и виброперемещение.

Для измерения параметров вибрации необходимо выбрать канал анализатора спектра, к которому подключен вибропреобразователь. В списке каналов доступны только те каналы анализатора спектра, размерностью которых является величина виброускорения: g или м/с<sup>2</sup> (1 g = 9,807 м/с<sup>2</sup> или 10 м/с<sup>2</sup> = 1,02 g). Величина виброускорения отображается в верхней строке в левой части окна программы (параметр "А" - acceleration - ускорение). Далее путем интегрирования виброускорения рассчитывается виброскорость и отображается во второй строке (параметр "V" - velocity - скорость). Постредством повторного интегрирования получается значение величины виброперемещения (параметр "S" - shift - перемещение).

Измеряемые значения могут быть как среднеквадратическими (СКЗ), так и пиковые. При измерении среднеквадратических значений возможно изменение времени усреднения - 0,1 с или 1 с.



В рамке "Пороги" пользователь устанавливает максимально допустимые пороговые значения по всем трем параметрам: виброускорению, виброскорости и виброперемещению. При установке галочки "Контроль" в реальном времени начинается контроль превышения текущих параметров вибрации над заданными пороговыми уровнями. Цветной индикатор информирует оператора о состоянии параметров вибрации. Зеленый цвет говорит о том, что параметр в норме. При и превышении пороговых уровней по любому из параметров цветной индикатор меняет цвет на красный. Дополнительно в систему подается глобальное сообщение, которое принуждает выключить выходной сигнал с генератора, если таковой

включен. Данная опция является еще одним фактором, обеспечивающим дополнительную безопасность оборудования при использовании, например, системы управления вибростендами при проведении испытаний изделий к устойчивости вибрации различного происхождения.

Программа может быть использована в качестве вибростопа для отключения оборудования при повышенном уровне вибрации.

При проведении измерений возможно применение общих частотных фильтров:

HP1 - для измерения виброускорения в частотном диапазоне от 1 до 20 000 Гц;

HP3 - для измерения виброускорения в частотном диапазоне от 3 до 20 000 Гц;

HP10 - для измерения виброускорения в частотном диапазоне от 10 до 20 000 Гц.

Эти фильтры служат для измерения виброускорения без коррекции. Фильтры HP1 и HP10 служат для удаления из результатов измерений низкочастотных компонент сигнала, связанных, зачастую, с паразитными наводками и искажениями.

Также возможно применение интегрирующих фильтров:

Vel1 - для измерения виброскорости в частотном диапазоне от 1 до 330 Гц;

Vel3 - для измерения виброскорости в частотном диапазоне от 3 до 1000 Гц;

Vel10 - для измерения виброскорости в частотном диапазоне от 10 до 3000 Гц;

Dil1 - для измерения виброперемещения в частотном диапазоне от 1 до 18 Гц;

Dil3 - для измерения виброперемещения в частотном диапазоне от 3 до 57 Гц;

Dil10 - для измерения виброперемещения в частотном диапазоне от 10 до 181 Гц;

VelMF (машинный фильтр) - для измерения виброскорости на стационарных машинах в частотном диапазоне от 10 до 1000 Гц (в соответствии с ISO 10816 и ГОСТ 25275 (СТ СЭВ 3173)).

Для измерения воздействия вибрации на организм человека возможно применение частотно-корректирующих фильтров:

W-Vz - для измерения общей вибрации (корректированного виброускорения) в вертикальной плоскости оси Z (в соответствии с ISO 8041 и СН 2.2.4/2.1.8.566-96);

W-Vxy - для измерения общей вибрации (корректированного виброускорения) в горизонтальной плоскости по осям XY (в соответствии с

ISO 8041 и СН 2.2.4/2.1.8.566-96);

W-Вс - для оценки воздействия вибрации на человека на спинке сидений (в соответствии с ISO 2631 и ISO 8041);

Wk - для измерения общей вибрации, воздействующей на человека по оси Z (в соответствии с ISO 8041, ISO 2631-1 и ГОСТ 12.1.012) (в настоящее время в России не применяются);

Wd - для измерения общей транспортной вибрации, воздействующей на человека по осям XY (в соответствии с ISO 8041, ISO 2631-1 и ГОСТ 12.1.012) (в настоящее время в России не применяются);

Wh - для измерения общей вибрации, воздействующей на человека (в соответствии с ISO 8041, ISO 2631-1 и ГОСТ 12.1.012);

Wс - для оценки воздействия вибрации на человека через спинку сидения (в соответствии с ISO 8041, ISO 2631-1 и ГОСТ 12.1.012) (в настоящее время в России не применяются);

Wj - для оценки воздействия вибрации на голову лежащего человека (в соответствии с ISO 8041, ISO 2631-1 и ГОСТ 12.1.012) (в настоящее время в России не применяются);

H-A - для измерения локальной вибрации (корректированного виброускорения), воздействующей на участки тела человека по трем осям XYZ (в соответствии с ISO 5349 и СН 2.2.4/2.1.8.566-96);

KB - для измерения общей вибрации на морских и речных судах ( $KB = W-Вс + 28,9$  дБ).

При аттестации рабочих мест и санитарно гигиенической оценке в настоящее время в России используются три типа фильтров: W-Вz, W-Вху, H-A.

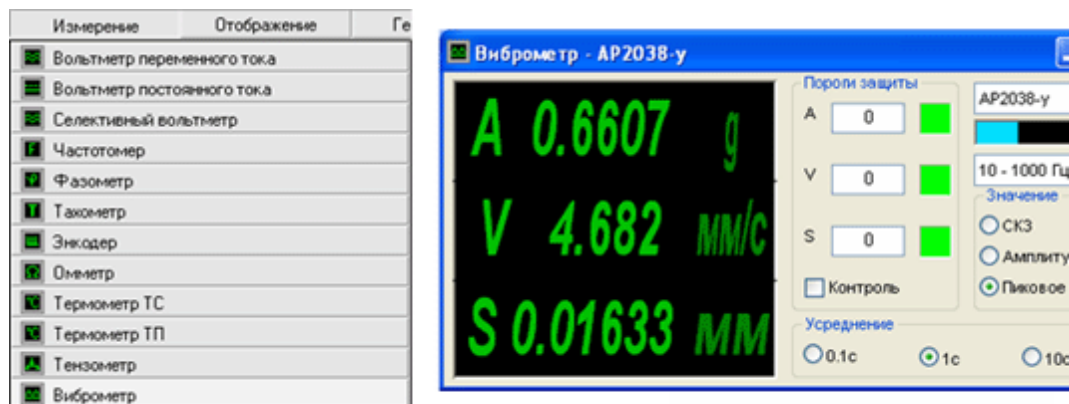
### О программе "Виброметр"

#### Назначение программы

Программа Виброметр предназначена для измерения среднеквадратических значений виброускорения, виброскорости и виброперемещения помощи пьезодатчиков (акселерометров), подключенных к входным к анализаторов сигнала. Программа Виброметр создает виртуальные мгновенных значений виброскорости и виброперемещения. С виброскорости и виброперемещения подвергаются фильтрации нижних частотой среза на уровне 3 дБ 1000 Гц и фильтрации верхних частот с частотой среза на уровне 3 дБ 10 Гц. Эти каналы доступны для последующего анализа в других программах ZETLab.

#### Описание программы

Для запуска программы Виброметр в меню Измерение панели ZETLab необходимо выбрать команду Виброметр. На экране монитора отобразится рабочее окно программы Виброметр. В заголовке окна программы будет отображаться название программы и название канала, по которому производятся измерения.



Примечание: программу Виброметр можно запустить непосредственно из директории ZETLab (по умолчанию: c:\ZETLab\). Имя запускаемого файла: Vbr.exe

В левой части рабочего окна программы Виброметр расположен графический индикатор, в котором отображаются значения виброускорения – напротив буквы A, виброскорости – напротив буквы V и виброперемещения напротив буквы S.

В рамке Пороги защиты устанавливаются максимально допустимые значения виброускорения, виброскорости и виброперемещения в полях справа от букв A, V и S соответственно. Значения вводятся с клавиатуры. Символом-разделителем для дробных чисел является точка.

Для включения режима вибростопа необходимо установить пороги и включить флажок Контроль. В этом режиме, при превышении заданных порогов ускорения, скорости или перемещения, индикатор, расположенный напротив соответствующего порога, меняет свой цвет с зеленого на красный и выдает глобальное сообщение, которое может использоваться, например, для автоматической остановки оборудования.

Справа, в верхнем поле со стрелкой (список) - выбирается канал ввода сигнала. В программе Виброметр доступны каналы с единицами измерения  $g$  или  $m/s^2$  и виртуальные (порожденные такими программами как ZETФормула, Фильтрация сигналов).

В полях со стрелками (списки) выбирать значения параметров можно следующими способами:

- нажать на стрелку поля и «мышью» выбрать из раскрывшегося списка нужное значение;
- нажать левой кнопкой «мыши» на поле и при помощи ролика «мыши» или кнопок клавиатуры со стрелками выбрать нужный элемент.

Индикатор Интегральный уровень сигнала показывает уровень и перегрузку. Если уровень сигнала превышает максимально допустимый уровень, индикатор становится полностью красного цвета, без черной правой части. Правая часть индикатора остается красным до тех пор, пока пользователь не нажмет на левую кнопку «мышь».

Ниже находится поле со стрелкой (список), в котором выбирается диапазон предварительной фильтрации сигнала.

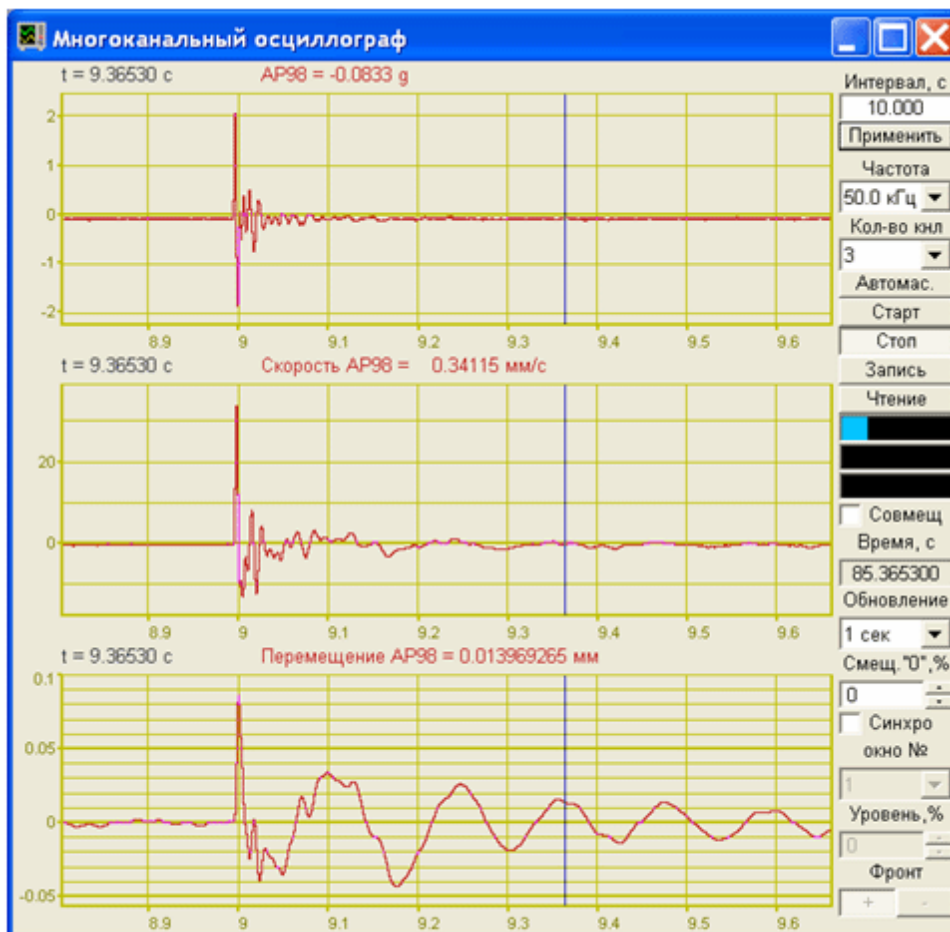
В группе Значение можно выбрать среднеквадратическое (СКЗ), амплитудное пиковое значение отображаемых величин.

В группе Усреднение можно выбрать усреднение 0.1 секунды, 1 секунда, 10 секунд.

Для выхода из программы необходимо нажать крестик в верхнем правом углу.

На рисунке ниже показана осциллограмма виброудара. На верхнем графике — ускорения, на среднем графике — сигнал виброскорости, на нижнем графике — виброперемещения.





### *Математическое описание ...*

Вибрация - это механические колебания тела.

Самый простой вид вибрации - это колебание или повторяющееся движение объекта около положения равновесия. Этот тип вибрации называется общей вибрацией, потому что тело перемещается как единое целое и все его части имеют одинаковую по величине и направлению скорость. Положением равновесия называют такое положение, в котором тело находится в состоянии покоя или положение которое оно займет, если сумма действующих на него сил равна нулю.

**СВОБОДНЫЕ КОЛЕБАНИЯ** — (собственные колебания), колебания в механич., электрич. или к. л. др. системе, совершающиеся при отсутствии внеш. воздействия за счёт первоначально внесённой энергии

(потенциальной или кинетической).

Вынужденные колебания — колебания, происходящие под воздействием внешних сил, меняющихся во времени.

Консервативный (физическая система, работа неконсервативных сил которой равна нулю и для которой имеет место закон сохранения механической энергии) гармонический осциллятор.

Второй закон Ньютона для такого осциллятора запишется в виде:

$$ma = -kx + F_0 \cos(\Omega t) . \text{ Если ввести обозначения:}$$

$$\omega_0^2 = \frac{k}{m}, \Phi_0 = \frac{F_0}{m},$$

и заменить ускорение на вторую производную от координаты по времени, то получим следующее обыкновенное дифференциальное уравнение:

$$\ddot{x} + \omega_0^2 x = \Phi_0 \cos(\Omega t) .$$

Решением этого уравнения будет сумма общего решения однородного уравнения и частного решения неоднородного. Общее решение однородного уравнения было уже получено здесь и оно имеет вид:

$$x(t) = A * \sin(\omega_0 t + \varphi),$$

где  $A, \varphi$  - произвольные постоянные, которые определяются из начальных условий.

Найдём частное решение. Для этого подставим в уравнение решение вида:  $x(t) = B * \cos(\Omega t)$  и получим значение для константы:

$$B = \frac{\Phi_0}{\omega_0^2 - \Omega^2} .$$

Тогда окончательное решение запишется в виде:

$$x(t) = A * \sin(\omega_0 t + \varphi) + \frac{\Phi_0}{\omega_0^2 - \Omega^2} \cos(\Omega t) .$$

Резонанс.

Из решения видно, что при частоте вынуждающей силы, равной частоте свободных колебаний, оно не пригодно — возникает резонанс, то есть

«неограниченный» линейный рост амплитуды со временем. Из курса математического анализа известно, что решение в этом случае надо искать в виде:

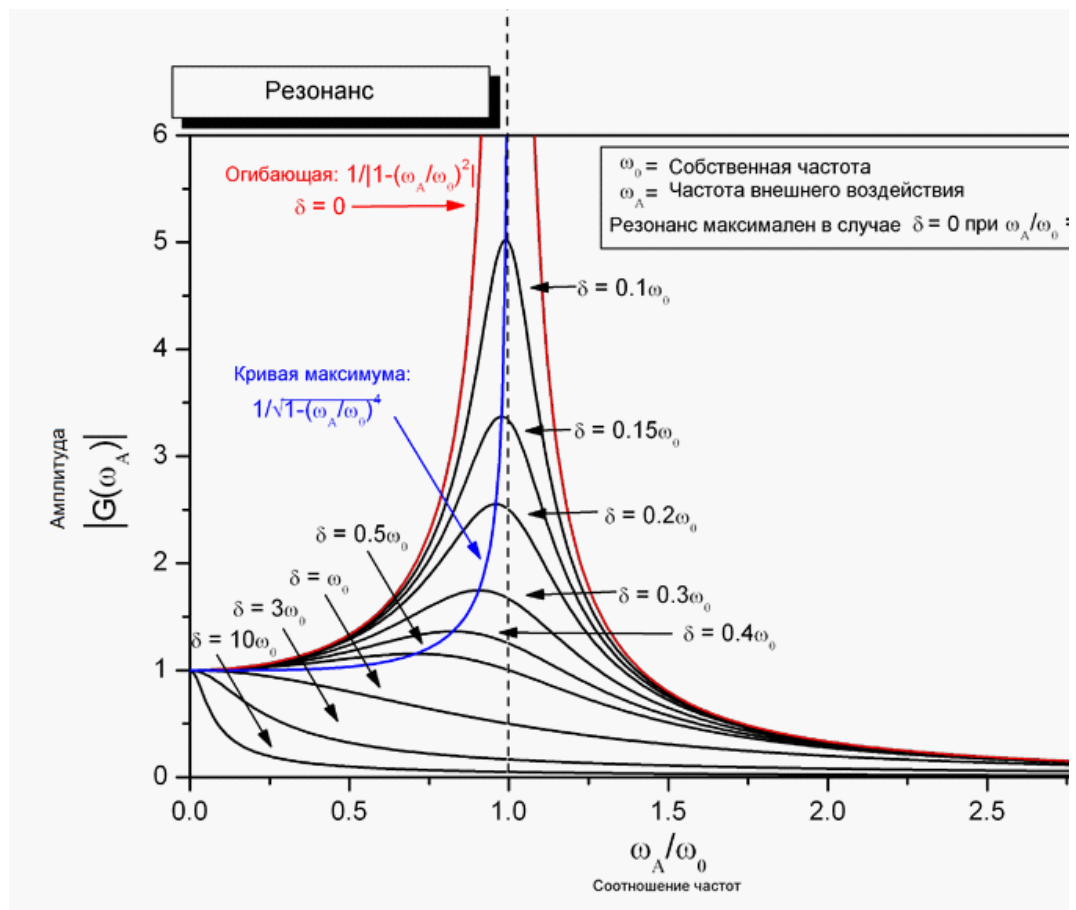
$$x(t) = T(A * \sin(\Omega t) + B * \cos(\Omega t))$$

Подставим часть выражения в дифференциальное уравнение и получим:

$$A = 0, B = \frac{\Phi_0}{2\Omega}$$

Таким образом, колебания в резонансе будут описываться следующим соотношением:

$$x(t) = \frac{\Phi_0}{2\Omega} * t * \sin(\Omega t)$$



Затухающий гармонический осциллятор.

Второй закон Ньютона:

$$m\ddot{x} = -kx - \alpha\dot{x} + F_0 \cos(\Omega t)$$

Переобозначения:

$$\omega_0^2 = \frac{k}{m}, \quad \Phi_0 = \frac{F_0}{m}, \quad \zeta = \frac{\alpha}{2\sqrt{km}}$$

Дифференциальное уравнение:

$$\ddot{x} + 2\zeta\omega_0\dot{x} + \omega_0^2x = \Phi_0 \cos(\Omega t)$$

Его решение будет строиться, как сумма решений однородного уравнения и частного решения неоднородного. Анализ однородного уравнения приведён здесь. Получим и проанализируем частное решение.

Запишем вынуждающую силу следующим образом:

$$\Phi_0 \cos \Omega t = \Phi_0 \operatorname{Re} e^{-i\Omega t}$$

тогда решение будем искать в виде:

$$x(t) = A * e^{-i\Omega t}, A \in \mathbb{C}$$

Подставим это решение в уравнение и найдём выражение для A:

$$A = \frac{\Phi_0}{\omega_0^2 - \Omega^2 - 2i\beta\Omega} = \frac{\Phi_0(\omega_0^2 - \Omega^2 + 2i\beta\Omega)}{(\omega_0^2 - \Omega^2)^2 + 4\beta^2\Omega^2} = |A| * e^{-i\varphi}$$

$$|A| = \frac{\Phi_0}{\sqrt{(\omega_0^2 - \Omega^2)^2 + 4\beta^2\Omega^2}}, \quad \varphi = -\arctan \frac{2\beta\Omega}{\omega_0^2 - \Omega^2}$$

где

Полное решение имеет вид:

$$x(t) = e^{-\zeta\omega_0 t} (C_1 \cos(\omega_d t) + C_2 \sin(\omega_d t)) + \operatorname{Re} \left[ \frac{\Phi_0(\omega_0^2 - \Omega^2 + 2i\beta\Omega)}{(\omega_0^2 - \Omega^2)^2 + 4\beta^2\Omega^2} e^{-i\Omega t} \right]$$

где  $\omega_d = \omega_0 \sqrt{1 - \zeta^2}$  - собственная частота затухающих колебаний.

Константы  $C_1$  и  $C_2$  в каждом случае определяются из начальных условий:

$$\begin{cases} x(0) = x_0 \\ \dot{x}(0) = v_0 \end{cases}$$

В этом случае, в отличие от осциллятора без трения, амплитуда колебаний в резонансе имеет конечную величину.

Если мы рассмотрим устоявшийся процесс, то есть ситуацию при  $t \rightarrow \infty$ , то решение однородного уравнения будет стремиться к нулю и останется только частное решение:

$$x(t \rightarrow \infty) = \Phi_0 \frac{(\omega_0^2 - \Omega^2) \cos \Omega t + 2\beta\Omega \sin \Omega t}{(\omega_0^2 - \Omega^2) + 4\beta^2 \Omega^2} = \frac{\Phi_0}{\sqrt{(\omega_0^2 - \Omega^2) + 4\zeta^2 \omega_0^2 \Omega^2}} \cos(\Omega t - \varphi)$$

Это означает, что при  $t \rightarrow \infty$  система “забывает” начальные условия, и характер колебаний зависит только от вынуждающей силы. Работа, совершаемая вынуждающей силой  $F(t) = F_0 \cos(\Omega t)$  за время  $dt$ , равна  $F dx$ , а

мощность  $P = F \frac{dx}{dt}$ . Из уравнения

$$\ddot{x} + 2\zeta\omega_0\dot{x} + \omega_0^2 x = \Phi_0 \cos(\Omega t),$$

Следует что

$$P(t) = F\dot{x} = (\ddot{x} + 2\zeta\omega_0\dot{x} + \omega_0^2 x)m\dot{x}$$

Если учесть, что при установившихся вынужденных колебаниях:

$$x = A \cos(\Omega t - \varphi)$$

$$\dot{x} = -A\Omega \sin(\Omega t - \varphi)$$

$$\ddot{x} = -A\Omega^2 \cos(\Omega t - \varphi)$$

То тогда средняя за период  $T = \frac{2\pi}{\Omega}$  мощность:

$$P = \frac{m}{T} \int_0^T (\ddot{x} + 2\zeta\omega_0\dot{x} + \omega_0^2 x)\dot{x} dt = A^2 m \zeta \omega_0 \Omega^2$$

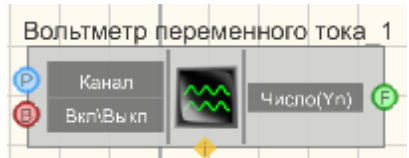
Работа за период:

$$A = m \int_0^T (\ddot{x} + 2\zeta\omega_0\dot{x} + \omega_0^2x)\dot{x} = A^2m\zeta\omega_0\Omega^2T$$

## 11.2. Вольтметр переменного тока

**Вольтметр переменного тока** - компонент. Служит для измерения уровня среднеквадратичного (истинного СКЗ) и пикового значений сигнала выбранного канала в установленных единицах измерения.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - входной канал, на который поступает сигнал, у которого нужно измерить требуемые величины.
- Вкл./Выкл - параметр, отвечающий за включение и выключение вольтметра.

#### ➤ Выходные

- Число (Уп) - выходной канал, с которого в потоке поступают уже измеренные значения входного сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- averagetime (0.1) - устанавливает время усреднения:
  1. 0.1 секунду (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 20 Гц).
  2. 1 секунду (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 2 Гц).

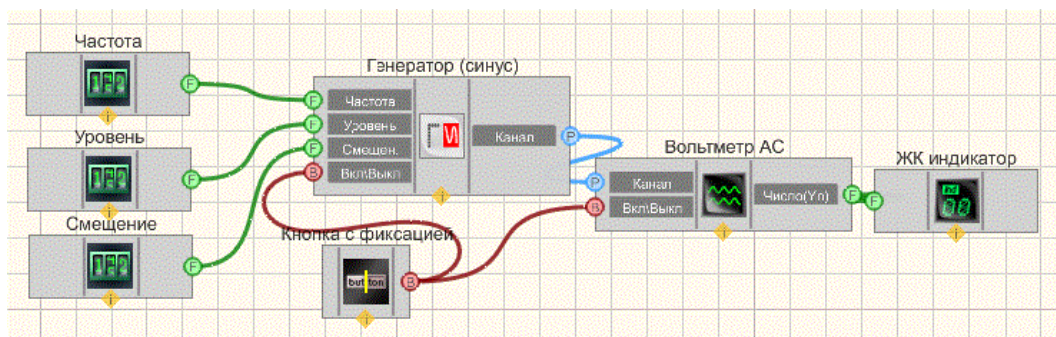
3. 10 секунд (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 0,2 Гц).

- **measuretype (СКЗ)** - тип измерения:
  1. СКЗ - (среднеквадратичное значение сигнала).
  2. Пик - максимальное значение на протяжении всего времени работы (значение пика  $\geq$  значения амплитуды).
  3. Амплитуда - максимальное значение смещения или изменения переменной величины от среднего значения.
- **Activate (true)** - состояние подачи сигнала (подан или отключен).
- **scale (Линейный)** - масштаб считывания данных:
  1. Линейный.
  2. Децибельный.



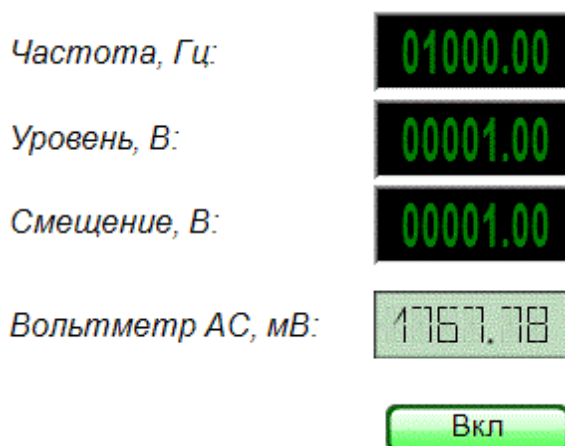
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Вольтметр переменного тока](#)<sup>[284]</sup> измеряет значение переменного напряжения, поступающего с [Синусоидального сигнала](#)<sup>[247]</sup>. [Селекторы](#)<sup>[213]</sup> служат для установки частоты, уровня и смещения генератора. Далее измеренное значение поступает на [ЖК индикатор](#)<sup>[348]</sup> для графического представления информации.

### Результат работы проекта



Данный пример непосредственно в ZETView

### **Внимание!**

Некоторые измерительные программы (вольтметр постоянного тока, виброметр, а также, в будущем, некоторые другие) при загрузке с помощью интерфейса UNIT (например, в ZETView) могут работать в многоканальном режиме. В этом режиме программа может выполнять свои измерения не по одному, а одновременно по нескольким каналам (количеством не более 128). При этом у каждого канала программы могут быть свои параметры, такие как время усреднения, частота дискретизации канала, тип используемого полосового фильтра для виброметра и т.д. Результаты измерений доступны родительской программе через UNIT.



### **Пример 2**

Данный пример непосредственно в ZETView



### **Математическое описание**

При описании переменных сигналов часто используются следующие величины.

**Среднее** или **среднеарифметическое значение** определяют так:

$$\bar{v} = \frac{1}{T} \int_{t_0}^{t_0+T} v(\tau) d\tau = \frac{1}{T} \int_0^T v(\tau) d\tau$$

Это значение равно площади под кривой временной функции, вычисляемой в



течение одного периода. Поскольку функция является периодической, среднеарифметическое значение  $v$  не зависит от начальной точки  $t_0$ . Для гармонических функций среднее значение  $v$  равно нулю.

**Среднее по модулю значение** определяется как:

$$\bar{v} = \frac{1}{T} \int_0^T |v(\tau)| dt$$

Это значение равной средней величине сигнала.

Примечание: среднее по модулю значение используется при вычислении величины заряда конденсатора, при работе с выпрямленными сигналами или в расчетах электролитических процессов. Выбор типа выпрямительных также основывается на среднем по модулю значении тока, поскольку падение на диоде практически всегда постоянно.

Для синусоидального напряжения справедливо:

$$\bar{v} = \frac{1}{T} \int_0^T \hat{v} |\sin \omega t| dt = \frac{2}{T} \hat{v} \int_0^{T/2} \sin \omega t dt = \frac{1}{\pi} \hat{v} [-\cos \omega t]_{\omega t=0}^{\omega t=\pi} = \frac{2}{\pi} \hat{v} \approx 0.637 \hat{v}$$

Аналогичное соотношение можно записать и для синусоидального тока.

**Среднеквадратичное значение (СКЗ)** напряжения характеризует мощность в цепи переменного тока. По определению СКЗ функции равно:

$$V_{\text{СКЗ}} = \sqrt{\frac{1}{T} \int_0^T v^2(t) dt}$$

Для синусоидального напряжения справедливо:

$$V_{\text{СКЗ}} = \sqrt{\frac{1}{T} \int_0^T (\hat{v} \sin \omega t)^2 dt} = \sqrt{\frac{1}{T} \hat{v}^2 \int_0^T \sin^2 \omega t dt} = \frac{\hat{v}}{\sqrt{2}} \approx 0,707 \hat{v}$$

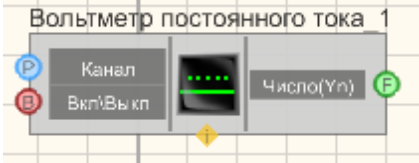
Аналогичное соотношение можно записать и для синусоидального тока.

Для переменных сигналов справедливо, что СКЗ значение всегда меньше или равно амплитудному значению.

## 11.3. Вольтметр постоянного тока

**Вольтметр постоянного тока** - компонент. Служит для измерения уровня постоянного сигнала выбранного канала в установленных единицах измерения.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - входной канал, на который поступает сигнал, у которого нужно измерить требуемые величины.
- Вкл\Выкл - параметр, отвечающий за включение и выключение вольтметра.

#### ➤ Выходные:

- Число (Уп) - выходной канал, с которого в потоке поступают уже измеренные значения входного сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

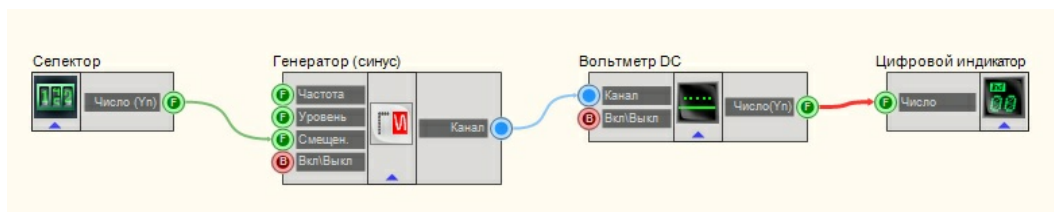
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- averagetime (0.1) - устанавливает время усреднения:
  1. 0.1 секунду.
  2. 1 секунду.
  3. 10 секунд.
- ReactionTime (0) - необходимая длительность работы сервера, по прошествии которой будет запущен вольтметр, с.
- Activate (true) - состояние подачи сигнала (подан или отключен).



### *Пример 1*

### *Проект в SCADA ZETView*



В этой схеме компонент [Вольтметр постоянного тока](#)<sup>[288]</sup> измеряет значение постоянной составляющей сигнала, поступающего с [Синусоидальный сигнал](#)<sup>[247]</sup>. [Селектор](#)<sup>[213]</sup> служат для установки значения смещения (измеряемой постоянной составляющей). Далее измеренное значение поступает на [Цифровой индикатор](#)<sup>[362]</sup> для графического представления информации.

### Результат работы проекта

Смещение сигнала, В:

00005.11

Показания вольтметра, мВ:

5109.91602

Данный пример непосредственно в ZETView

### Внимание!

Некоторые измерительные программы (вольтметр постоянного тока, виброметр, а также, в будущем, некоторые другие) при загрузке с помощью интерфейса UNIT (например, в ZETView) могут работать в многоканальном режиме. В этом режиме программа может выполнять свои измерения не по одному, а одновременно по нескольким каналам (количеством не более 128). При этом у каждого канала программы могут быть свои параметры, такие как время усреднения, частота дискретизации канала, тип используемого полосового фильтра для виброметра и т.д. Результаты измерений доступны родительской программе через UNIT.



### Пример 2

Данный пример непосредственно в ZETView



### Математическое описание

Напряжение (разность потенциалов) между точками А и В – отношение работы электрического поля при переносе электрического заряда из точки А в точку В к величине пробного заряда.

$$\varphi_A - \varphi_B = U_{AB} = \frac{A_{q:A \rightarrow B}^{\text{поле}}}{q}$$

При этом считается, что перенос пробного заряда не изменяет распределения зарядов на источниках поля (по определению пробного заряда).

Альтернативное определение (для электростатического поля):

$$\varphi_A - \varphi_B = U_{AB} = \int_A^B (\vec{E}, d\vec{l})$$

Это интеграл от проекции поля (напряжённости поля) на расстояние между точками А и В вдоль любой траектории, идущей из точки А в точку В.

## 11.4. Диапазон значений

**Диапазон значений** - компонент. Компонент предназначен для получения мгновенных значений сигнала канала в заданном диапазоне времени.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

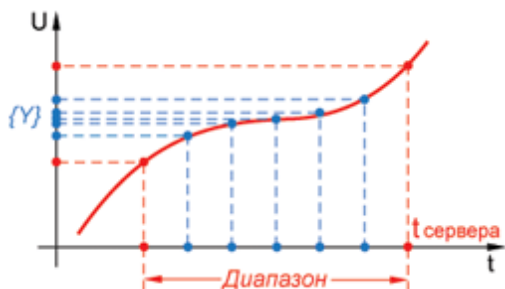
➤ Входящие:

- Канал - вывод, к которому подключается канал, в котором требуется получить массив мгновенных значений.
- Время(с) - предназначен для получения значения времени сервера. Данное значение является последней точкой диапазона передаваемых значений (t сервера на рисунке ниже). Компонент срабатывает при получении значения данным контактом.
- Диапазон(с) - вывод для получения численного значения интервала времени, в

течении которого будут сниматься значения.

➤ Выходные

- Канал Y - массив мгновенных значений анализируемого сигнала в указанном диапазоне.
- Канал X - массив индексов элементов массива {Y}.



**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

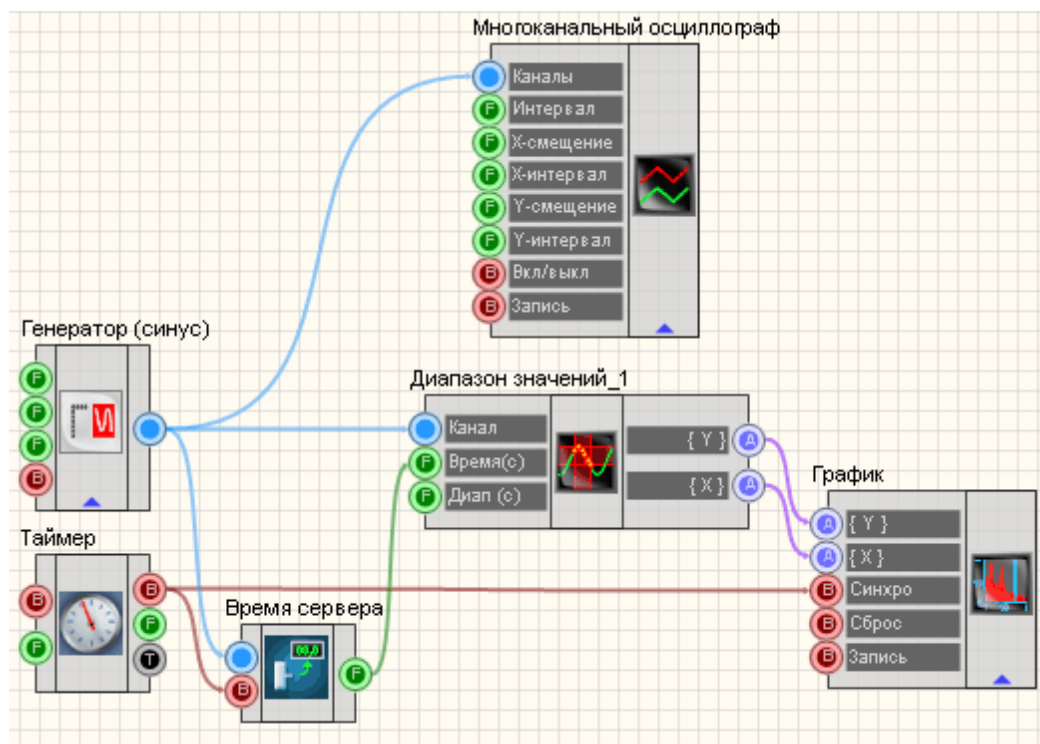
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Interval (1) - установка значения временного интервала, с.
- freqrange (50000) - установка диапазона частот, Гц.



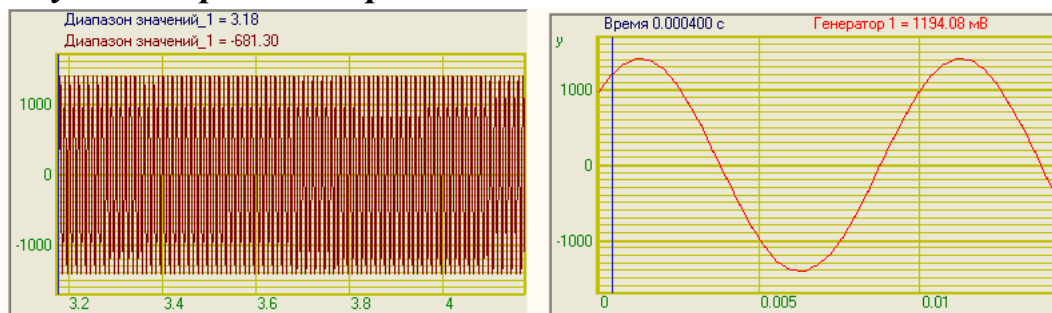
**Пример**

**Проект в SCADA ZETView**



В этой схеме компонент [Диапазон значений](#)<sup>[290]</sup> предназначен для получения мгновенных значений сигнала канала в заданном диапазоне времени. [Таймер](#)<sup>[183]</sup> служит для подачи синхронизирующих импульсов на [Время сервера](#)<sup>[180]</sup> и [График](#)<sup>[490]</sup>. Это нужно, чтобы [График](#)<sup>[490]</sup> и [Диапазон значений](#)<sup>[290]</sup> работали в едином временном пространстве. [Синусоидальный сигнал](#)<sup>[247]</sup> нужен для создания пробного сигнала, а [Многоканальный осциллограф](#)<sup>[494]</sup> - для его текущего графического отображения.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 11.5. Мгновенное значение

**Мгновенное значение** - компонент. Предназначен для считывания мгновенного значения сигнала в канале в указанный момент времени.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Канал - вывод, к которому подключается канал, в котором будут определяться мгновенные значения.
- Время - вывод, к которому подключается канал со значениями моментов времени, мгновенное значение в которых требуется определить.

## ➤ Выходные

- Число(Yn) - вывод, с которого идет получение мгновенного значения сигнала канала в требуемый момент времени.

**Настраиваемые свойства:**

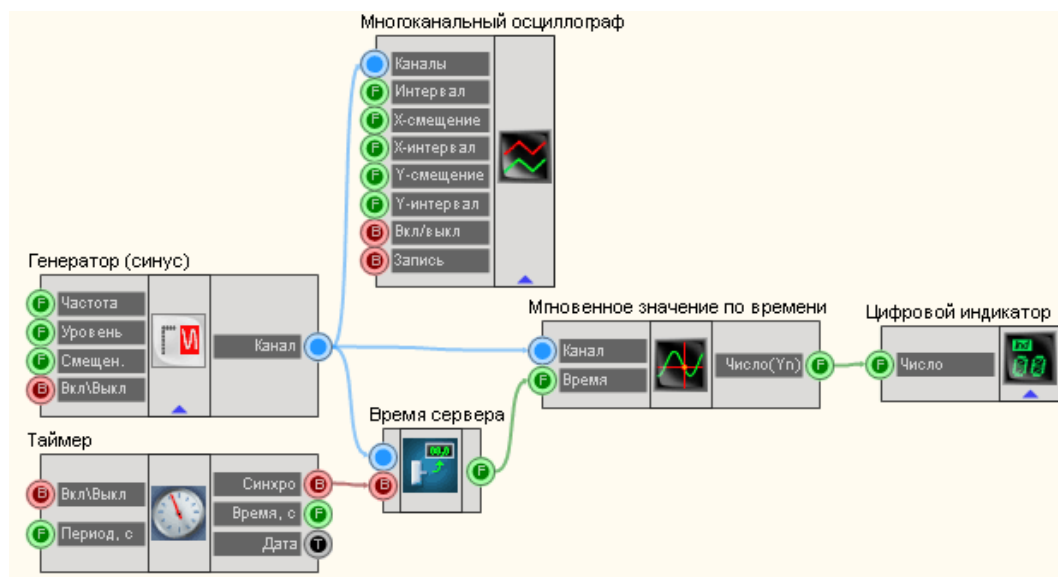
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- TimeWait (0.2) - допустимое время рассинхронизации, сек. Измеряется в сек, по умолчанию равно 0,2 сек. Если время по каналу отстаёт от запрошенного времени (что может быть у виртуальных каналов), то тогда компонент будет ждать некоторое время.

***Пример******Проект в SCADA ZETView***



В этой схеме компонент [Синусоидальный сигнал](#)<sup>[247]</sup> выдает синусоидальный сигнал. [Многоканальный осциллограф](#)<sup>[494]</sup> нужен для графического представления сигнала. Компонент [Таймер](#)<sup>[183]</sup> генерирует каждые 0,3 секунды синхроимпульсы, которые поступают на вход [Время сервера](#)<sup>[180]</sup>. [Цифровой индикатор](#)<sup>[362]</sup> нужен для отображения численной информации. На вход Мгновенное значение по времени поступает информация канала и время с [Время сервера](#)<sup>[180]</sup> (с периодичностью один раз в 0,3 секунды, при этом каждый раз в данный момент времени Мгновенное значение по времени отправляет численное значение сигнала).

### Результат работы проекта





Данный пример непосредственно в ZETView

## 11.6. Параметры дисбаланса

<Раздел находится в разработке>

**Параметры дисбаланса** - компонент. Предназначен для определения параметров дисбаланса.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входящие:

- Канал ДО - вывод, к которому подключается канал содержащий имя канала датчика оборотов.
- Канал А1 - вывод, к которому подключается канал с именем акселерометра №1.
- Вкл\Выкл - параметр, отвечающий за включение и выключение параметров дисбаланса.

➤ Выходные

- Скорость - вывод, с которого идет получение скорости вращения, об/мин.
- Дисбаланс - величина дисбаланса, г\*см.
- Угол Max - угол максимума, град. от 0 до 360
- Угол Min - угол минимума, град. от 0 до 360

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

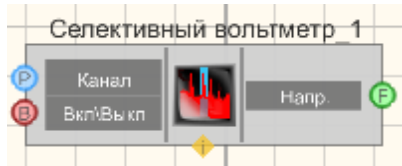
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- timeaverage (1.0)- время усреднения, с.
- FreqBalance (12000) - номинальная частота балансировки, об/мин.(частота в Гц должна быть кратной частоте дискретизации АЦП в Гц).
- Activate (1) - состояние работы компонента
- MultiFactor (0.000130) - устанавливается значение мультиплекативного множителя.
- Degrefactor - (3,5) - устанавливается значение показателя степени.

## 11.7.Селективный вольтметр

**Селективный вольтметр** - компонент. Предназначен для измерения среднеквадратичного (СКЗ, True RMS) и пикового (пик-пик) значения напряжения переменного тока на основной (несущей) частоте сигнала. Особенностью селективного вольтметра является исключение влияния гармоник на показания.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - входной канал, на который поступает сигнал, у которого нужно измерить требуемые величины.
- Вкл/Выкл - входной контакт, отвечающий за включение и выключение вольтметра.

#### ➤ Выходные

- Напряжение - выходной канал, с которого в потоке поступают уже измеренные значения входного сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

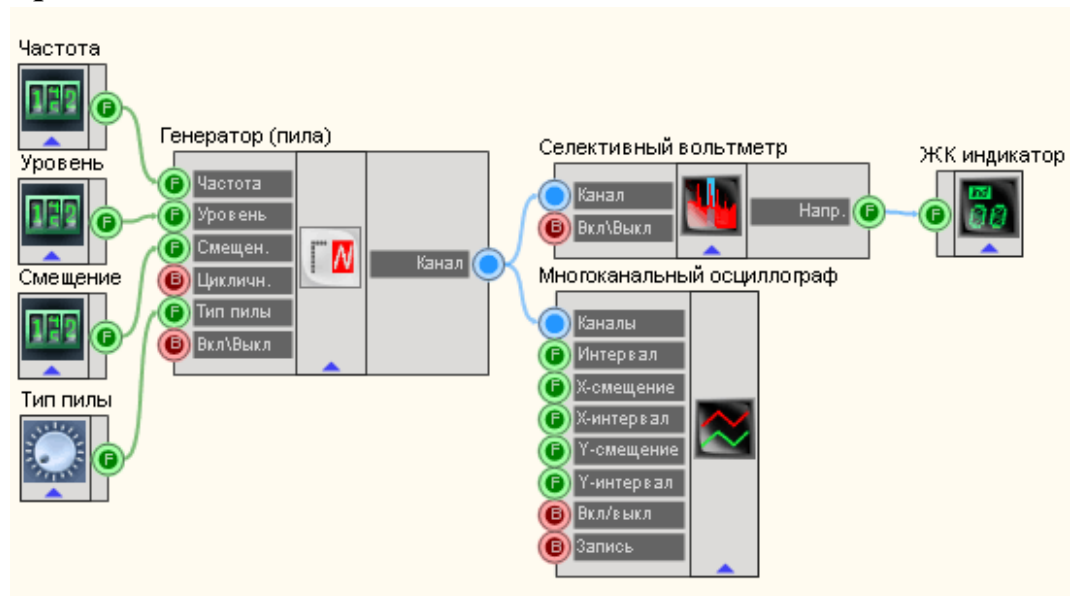
- averagetime (0.1) - устанавливает время усреднения:
  1. 0.1 секунду (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 20 Гц).
  2. 1 секунду (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 2 Гц).
  3. 10 секунд (при этом СКЗ и пиковое значение правильно измеряются для сигнала с частотой не менее 0,2 Гц).
- AutoFrequency (false) - автоматическая установка вольтметра на измеряемую частоту.
- Frequency (0) - частота, на которой вольтметр будет делать измерения, Гц.

- FrequencyBar (0) - полоса частот, на которых вольтметр будет делать измерения, Гц.
- Activate (true) - состояние подачи сигнала (подан или отключен).



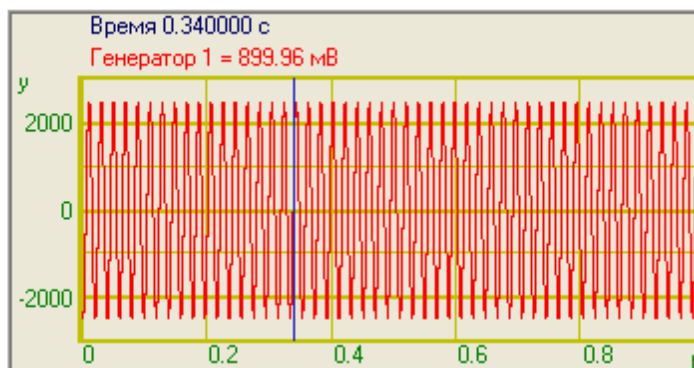
## Пример

### Проект в SCADA ZETView



В приведенном проекте компонент [Селективный вольтметр](#)<sup>[296]</sup> измеряет среднее квадратичное значение пилообразного сигнала, поступающего с [Пилообразный сигнал](#)<sup>[239]</sup>. [Селекторы](#)<sup>[213]</sup> (также [другого типа](#)<sup>[217]</sup>) служат для установки значений частоты, уровня, смещения и типа пилообразного сигнала. [Многоканальный осциллограф](#)<sup>[494]</sup> нужен для визуального восприятия сигнала. [ЖК индикатор](#)<sup>[348]</sup> служит для графического представления значения СКЗ пилообразного сигнала.

### Результат работы проекта



Частота, Гц:                      Уровень, В:                      Смещение, В

00010.00

00001.00

00003.00



Показания  
селективного  
вольтметра, мВ:

1764.92

Данный пример непосредственно в ZETView

## F

### Математическое описание

При описании переменных сигналов часто используются следующие величины.

**Среднее** или **среднеарифметическое значение** определяют так:

$$\bar{v} = \frac{1}{T} \int_{t_0}^{t_0+T} v(\tau) d\tau = \frac{1}{T} \int_0^T v(\tau) d\tau$$

Это значение равно площади под кривой временной функции, вычисляемой в течение одного периода. Поскольку функция является периодической, среднеарифметическое значение  $v$  не зависит от начальной точки  $t_0$ . Для гармонических функций среднее значение  $v$  равно нулю.

**Среднее по модулю значение** определяется как:

$$\bar{v} = \frac{1}{T} \int_0^T |v(\tau)| d\tau$$

Это значение равной средней величине сигнала.

Примечание: среднее по модулю значение используется при вычислении величины заряда конденсатора, при работе с выпрямленными сигналами или в расчетах электролитических процессов. Выбор типа выпрямительных также основывается на среднем по модулю значении тока, поскольку падение на диоде практически всегда постоянно.

Для синусоидального напряжения справедливо:

$$\bar{v} = \frac{1}{T} \int_0^T \hat{v} |\sin \omega t| dt = \frac{2}{T} \hat{v} \int_0^{T/2} \sin \omega t dt = \frac{1}{\pi} \hat{v} [-\cos \omega t]_{\omega t=0}^{\omega t=\pi} = \frac{2}{\pi} \hat{v} \approx 0,637 \hat{v}$$

Аналогичное соотношение можно записать и для синусоидального тока.

**Среднеквадратичное значение (СКЗ)** напряжения характеризует мощность в цепи переменного тока. По определению СКЗ функции равно:

$$V_{\text{СКЗ}} = \sqrt{\frac{1}{T} \int_0^T v^2(t) dt}$$

Для синусоидального напряжения справедливо:

$$V_{\text{СКЗ}} = \sqrt{\frac{1}{T} \int_0^T (\hat{v} \sin \omega t)^2 dt} = \sqrt{\frac{1}{T} \hat{v}^2 \int_0^T \sin^2 \omega t dt} = \frac{\hat{v}}{\sqrt{2}} \approx 0,707 \hat{v}$$

Аналогичное соотношение можно записать и для синусоидального тока.

Для переменных сигналов справедливо, что СКЗ значение всегда меньше или равно амплитудному значению.

## 11.8. Тахометр

**Тахометр** - компонент. Средство измерения частоты вращения (числа оборотов в единицу времени) деталей машин и механизмов.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - контакт, к которому подключается измерительный канал.
- Сброс - при подаче управляющего импульса на этот контакт, происходит сброс накопленного числа оборотов в ноль.
- Вкл/Выкл - параметр, отвечающий за включение и выключение тахометра.

#### ➤ Выходные

- Скорость - измеренное значение скорости вращения, обр/мин.
- Обороты - количество совершенных оборотов с момента последнего сброса.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- DriveParametr1 (1) - кинематический параметр ведущего вала 1.
- DriveParametr2 (1) - кинематический параметр ведущего вала 2.
- DriveParametr3 (1) - кинематический параметр ведущего вала 3.
- DrivenParametr1 (1) - кинематический параметр ведомого вала 1.
- DrivenParametr2 (1) - кинематический параметр ведомого вала 2.
- DrivenParametr3 (1) - кинематический параметр ведомого вала 3.
- AutoLevel (false) - автоматическая установка порога.
- Lowlevel (0) - нижний порог (в единицах измерения).
- HighLevel (0) - верхний порог (в единицах измерения).
- Activate (true) - состояние работы тахометра.
- RPS (0) - установка размерности частоты вращения (0 - об/мин, 1 - об/сек).
- VirtChanRevs (0) - создание виртуального канала кол-ва оборотов (1 - создавать, 0 - нет)
- Amplify (1) - установка множителя (1, 10, 100, 1000, 10000, 100000).

## 11.9.Текущий уровень

**Текущий уровень** - компонент. Средство измерения текущего уровня сигнала по каналам.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Канал - контакт, к которому подключается измерительный канал.

## ➤ Выходные

- Уровень - измеренное значение текущего уровня.

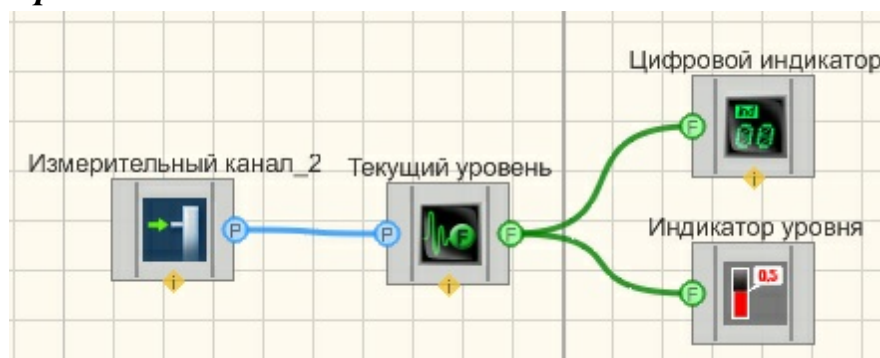
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

- Не имеет.

**Пример****Проект в SCADA ZETView****Результат работы проекта**



## 11.10.Тензометр

**Тензометр** - компонент. Предназначен для работы с данными, поступающими с канала тензодатчика. Служит для получения механических напряжений в конструкциях.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - контакт, к которому подключается измерительный канал.
- Опора - канал, к которому подключается питание измерительного моста.
- Сброс - значение, в которое сбрасывается текущее показание.
- Сброс - при подаче на этот контакт происходит сброс значения.
- Вкл\Выкл - параметр, отвечающий за включение и выключение тензометра.

#### ➤ Выходные

- Число(Уп) - контакт, с которого поступают выходные значения тензометра.
- Канал - виртуальный выходной канал, с которого поступают мгновенные значения, получаемые с тензометра.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Unit - указание единиц измерения.
- measuretype (Абсолютные) - тип измерений:
  1. Абсолютные.



2. Относительные (значение единиц относительно опорного канала).
- powertype (Постоянный ток) - тип питания датчика:
    1. Постоянный ток.
    2. Переменный ток - расчет значений будет проводиться по алгоритму селективного вольтметра, и, соответственно, измерительная схема (первичный преобразователь) должна быть запитана переменным током.
  - Smooth (10) - сглаживание (время в [мс], за которое будет проводиться усреднение измеренных значений)
  - Inversion (false) - отвечает за изменение знака измерений (позволяет инвертировать режим работы тензодатчика: сжатие-растяжение и наоборот).
  - mode (Тензорезистор) - режим работы датчика:
    1. Тензорезистор - в этом режиме.
    2. Тензодатчик - в этом режиме нужно ввести значение Sensitivity (чувствительности, мВ/В) и Limit (предел измерения в единицах Unit).
  - Sensitivity (1) - чувствительность тензодатчика, мВ/В.
  - Limit (10) - предел измерения тензодатчика.
  - UseClbFile (false) - разрешение использования калибровочного файла, в котором находятся сохраненные ранее настройки и калибровочные таблицы для тензорезистора.
  - ClbFileName - указывается путь к калибровочному файлу тензорезистора.
  - Value (0) - число, в которое сбрасывается текущее показание.
  - Activate (true) - состояние работы тензометра.



### **Пример**

Более подробно об использовании Тензометра можно ознакомиться на официальном сайте, по ссылкам:

1. [Ссылка №1.](#)
2. [Ссылка №2.](#)



### **Математическое описание**

В основе работы тензорезистивных преобразователей лежит свойство материалов изменять свое электрическое сопротивление при механических деформациях под действием приложенной силы. Конструктивно большинство тензорезисторов (рис. 1) выпускается в виде проводников, жестко связанных с бумажной или пленочной основой 2. Проводник 3 представляет собой так называемую решетку из зигзагообразно уложенной тонкой проволоки диаметром 0,02-0,05 мм, к концам которой пайкой или

сваркой присоединяются выводные медные проводники 4. Сверху проводники закрываются бумагой или пленкой или покрываются лаком 1. После наклеивания подложки тензорезистора на поверхность деформация этой поверхности передается проводниками и приводит к изменению их сопротивления.

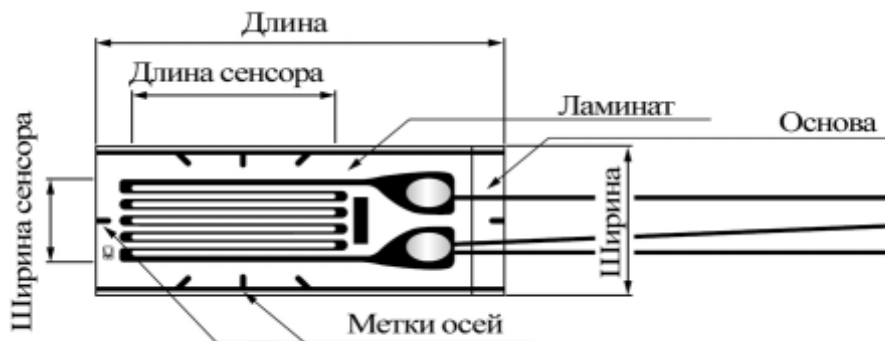


Рисунок 1.

Как видно на рисунке 1 датчик состоит из основы, на которую нанесен проводящий слой, образующий змейку, на “поворотах” толщина проводника увеличена для снижения чувствительности на растяжение перпендикулярной основной оси. Основная ось проходит вдоль линий направления змейки (на рисунке по горизонтали) и растяжение датчика вдоль этого направления вызывает максимальное изменение сопротивления датчика. Сверху датчик покрыт слоем прозрачного ламината, который защищает резистивный слой от повреждений. На датчике также нанесены метки, указывающие направление осей, упрощающих его установку, обычно проходят через центр датчика под углами  $90^\circ, \pm 45^\circ$  относительно основной оси чувствительности. При построении системы контроля нагрузки на различные механические конструкции обычно применяется несколько тензодатчиков или системы датчиков, оси которых пересекаются под различными углами, либо идут параллельно, поэтому для упрощения монтажа обычно используют датчики, имеющие несколько тензорезистивных элементов и уже расположенных под определенным углом.

Известно, что под действием силы или системы сил на металлический предмет он начинает деформироваться, причем до точки предела текучести материала деформация носит упругий характер и подчиняется закону Гука. Это означает, что при снятии силы предмет принимает исходные размеры, а величина напряжения равна линейной

деформации умноженной на модуль Юнга. Формулы для расчета напряжения и деформации приведены ниже:

$$\varepsilon = \frac{\Delta L}{L};$$

$$\sigma = \frac{F}{S};$$

$$\delta = E * \varepsilon;$$

где  $\varepsilon$  – линейная деформация,  $E$  – модуль Юнга,  $\sigma$  – напряжение,  $F$  – сила приведшая к деформации,  $S$  – площадь поперечного сечения, на который эта сила действует,  $L$  – исходная длина,  $\Delta$  – изменение длины под действием силы. Существует также понятия продольной и поперечной деформации. Продольная деформация – это деформация тела вдоль линии действия силы. Поперечная – деформация тела относительно оси, перпендикулярной направлению действия силы. Их отношения называются коэффициентом Пуассона:

$$\nu = \left| \frac{\varepsilon_{\text{поп}}}{\varepsilon_{\text{продол}}} \right|$$

Для тензорезистивных датчиков основная формула выглядит следующим образом:

$$\frac{\Delta R}{R} = K\varepsilon$$

где  $\Delta R$  – изменение сопротивления тензодатчика вызванное деформацией  $\varepsilon$ ,  $K$  – коэффициент тензочувствительности датчика (табличная величина),  $R$  – исходное сопротивление. Если подробнее расписать это выражение, то получим:

$$\Delta R = \frac{RKF}{SE}$$

Из формулы вытекает, что изменение сопротивления тензодатчика, установленного на металлическую балку, поперечного сечения  $S$ , прямо пропорционально площади поперечного сечения балки и модулю Юнга. Измерив изменение сопротивления тензодатчика, можно рассчитать нагрузку на рассматриваемую балку, зная ее поперечное сечение и свойство материала из которого она изготовлена. Эта формула верна только для случаев растяжения/сжатия.

Для ситуации изгиба весь процесс выглядит несколько иначе, поскольку деформация рассчитывается сложнее. Нужно учитывать форму поперечного сечения балки и расстояние от места приложения силы к центру тензодатчика. Деформация в этом случае может быть рассчитана следующим образом:

$$\varepsilon = \frac{M}{ZE},$$

где  $M$  – момент силы,  $Z$  – момент сопротивления сечения. Для различных вариантов размещения тензодатчика, крепления балки, и местоположения точки приложения силы рассчитывается различными способами.

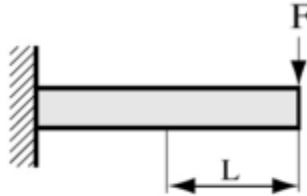
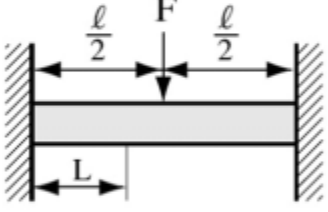
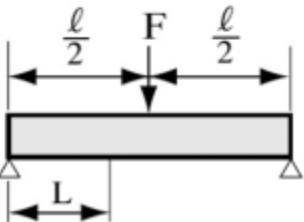
Схема приложения силы	Формула вычисления
	$M = FL$
	$L = \frac{l}{2}; \Rightarrow M = \frac{Wl}{8}$
	$L = \frac{l}{2}; \Rightarrow M = \frac{Wl}{4}$

Таблица 1. Расчет момента силы для различных вариантов положения балки и тензодатчика.

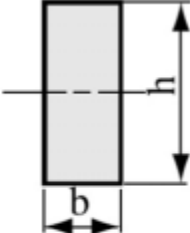
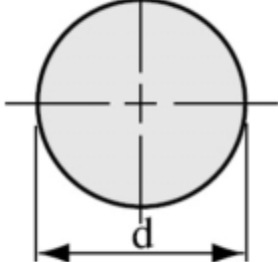
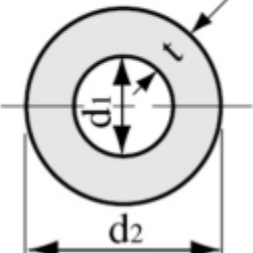
Поперечное сечение	Формула расчета
	$\frac{1}{6}bh^2$
	$\frac{\pi}{32}d^3$
	$\frac{\pi (d_2^4 - d_1^4)}{32d_2}$

Таблица 2. Расчет момента сопротивления сечения для различной конфигурации сечения.

Для мониторинга крутящего момента используются тензодатчики, расположенные на балке под углом  $45^\circ$  относительно оси вращения рисунок 2. Используется один, два или четыре датчика. При установке двух датчиков их оси располагаются под углом  $90^\circ$  относительно друг друга и под углом  $45^\circ$  и  $-45^\circ$  относительно оси вращения. При этом один из датчиков испытывает деформацию сжатия, другой – растяжения. При использовании четырех тензорезисторов они располагаются “крестиком”.

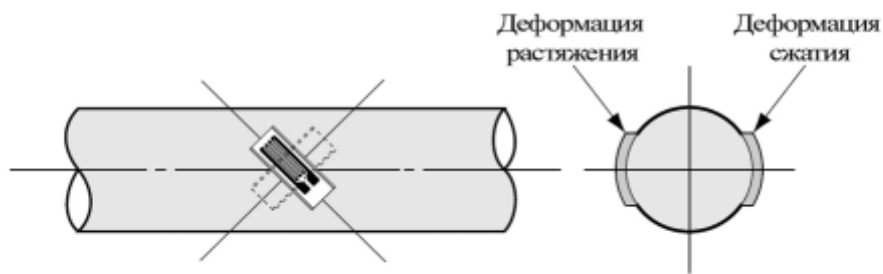


Рисунок 2.

Вращающее усилие может быть рассчитано следующим образом:

$$T = \frac{\varepsilon * E * Z_p}{1 + \nu}$$

где  $\nu$  – коэффициент Пуассона,  $T$  – вращающий момент,  $E$  – модуль Юнга,  $Z_p$  – угловое сопротивление сечения. Для цельного цилиндрического стержня:

$$Z_p = \frac{\pi d^3}{16}$$

Для цилиндрической полой трубы с внешним диаметром  $d_2$  и внутренним  $d_1$ :

$$Z_p = \frac{\pi(d_2^4 - d_1^4)}{32d_2}$$

Поскольку  $\Delta R$  очень малая величина по сравнению с  $R$ , то измерение ее производят, включив датчик в схему моста Уитстона.

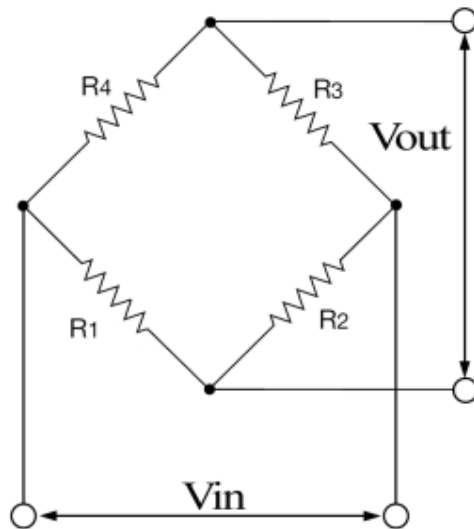


Рисунок 3.

Принцип измерения основан на взаимной компенсации сопротивлений двух звеньев, одно из которых включает измеряемое сопротивление. В качестве индикатора обычно используется чувствительный гальванометр, показания которого должны быть равны нулю в момент равновесия моста. На схеме  $R_1$ ,

$R_2$ ,  $R_3$ ,  $R_4$  – плечи моста.  $R_3$  представляет собой неизвестное сопротивление;  $R_1$ ,  $R_2$  и  $R_4$  — известные сопротивления, причём значение  $R_2$  может регулироваться. Если отношение сопротивлений ( $R_2/R_1$ ) равно отношению сопротивлений другого ( $R_3/R_4$ ), то разность потенциалов между двумя средними точками будет равна нулю, и ток между ними не будет протекать. Сопротивление  $R_2$  регулируется до получения равновесия, а направление протекания тока показывает, в какую сторону нужно регулировать  $R_2$ .

С помощью гальванометра момент равновесия можно установить с большой точностью, и если сопротивления  $R_1$ ,  $R_2$  и  $R_4$  имеют маленькую погрешность, то  $R_3$  может быть измерено очень точно, ведь даже небольшие изменения  $R_x$  вызывают заметное нарушение баланса всего моста. Когда мост сбалансирован:

$$R_3 = \frac{R_2 R_4}{R_1}$$

Если известны значения всех четырёх сопротивлений, а также напряжение ( $V_{in}$ ), то напряжение на плечах моста можно найти, используя формулы делителя напряжения, а затем вычесть их друг из друга, чтобы найти  $V_{out}$ :

$$V_{out} = \left( \frac{R_3}{R_4 + R_3} - \frac{R_2}{R_1 + R_2} \right) V_{in} = \frac{R_3 R_1 - R_4 R_2}{(R_4 + R_3)(R_1 + R_2)} V_{in}$$

Если наш тензорезистор ( $R_3$ ) будет иметь сопротивление  $R + \Delta R$ , а остальные резисторы равны  $R$ , то имеем:

$$V_{out} = \frac{(R + \Delta R)R - RR}{(R + R + \Delta R)(R + R)} V_{in} = \frac{\Delta R R}{2R(2R + \Delta R)} V_{in} = \frac{\Delta R}{4R + 2\Delta R} V_{in}$$

так как  $\Delta R$  очень малая величина по сравнению с  $R$ , то напряжение на выходе будет равно:

$$V_{out} = \frac{1}{4} \frac{\Delta R}{R} V_{in} = \frac{1}{4} K \varepsilon E$$

При решении задач измерения нагрузки на те или иные конструкции обычно

в измерительный мост включается один, два или четыре тензорезистора.

Первый и главнейший фактор – температура, а вернее ее влияние не только на сопротивление самого тензодатчика. Но и на линейные размеры контролируемого объекта. Существует 2 способа решения этой проблемы:

- Использование термокомпенсированных тензодатчиков.
- Включение в измерительный мост дополнительного тензодатчика.

Рассмотрим второй способ. Для компенсации температурного влияния на баланс самого измерительного моста часто используют дополнительный тензодатчик, включенный в мост таким образом, чтобы изменение его сопротивления, обусловленное дрейфом температуры, было согласовано с изменением сопротивления основного тензодатчика. При этом дополнительный датчик монтируется на балке, к которой не прикладывается усилие, но находящейся под той же температурой, что и контролируемая конструкция. Например для компенсации тензодатчика, включенного как  $R_1$  дополнительный тензодатчик ставится на место  $R_4$ . В этом случае изменение сопротивления, обусловленное изменением температуры на основном тензодатчике, равно изменению сопротивления в дополнительном тензодатчике и баланс измерительного моста сохраняется.

При построении измерительного моста на практике обычно оказывается, что сами датчики находятся на довольно большом расстоянии друг от друга. При этом длина провода может составлять несколько сотен метров. При этом нужно будет также учитывать температурный эффект в проводах.

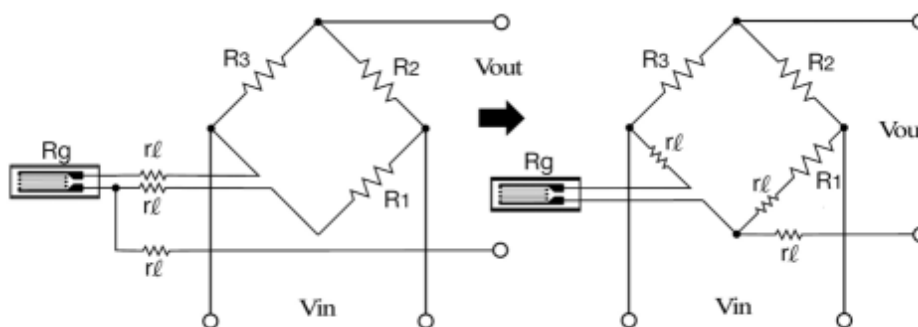


Рисунок 4.

На схеме приведена схема компенсации. Как видно из рисунка,



сопротивление двух проводов входят в измерительный мост и компенсируют друг друга, сопротивление третьего соединяют последовательно с выходным напряжением. При использовании измерителя с входным сопротивлением много большим сопротивления резисторов можно не учитывать.

## 11.11.Термометр ТП

**Термометр термопар** - компонент. Предназначен для работы с данными, поступающими с канала термопары. Служит для преобразования термоЭДС в численное значение температуры.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤Входные:

- Канал - вывод, к которому подключается канал с термопарой.
- КХС - компенсатор холодного спая - канал служащий для компенсации величины паразитной термоЭДС, возникающей при подключении термопары к клеммам.
- Вкл\Выкл - параметр, отвечающий за включение и выключение термометра термопар.

#### ➤Выходные

- Число(T) - вывод, с которого поступает усредненное за 0,1 сек значение температуры.
- Канал - виртуальный выходной канал, с которого поступают мгновенные значения температуры.

### Настраиваемые свойства:

#### ➤Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

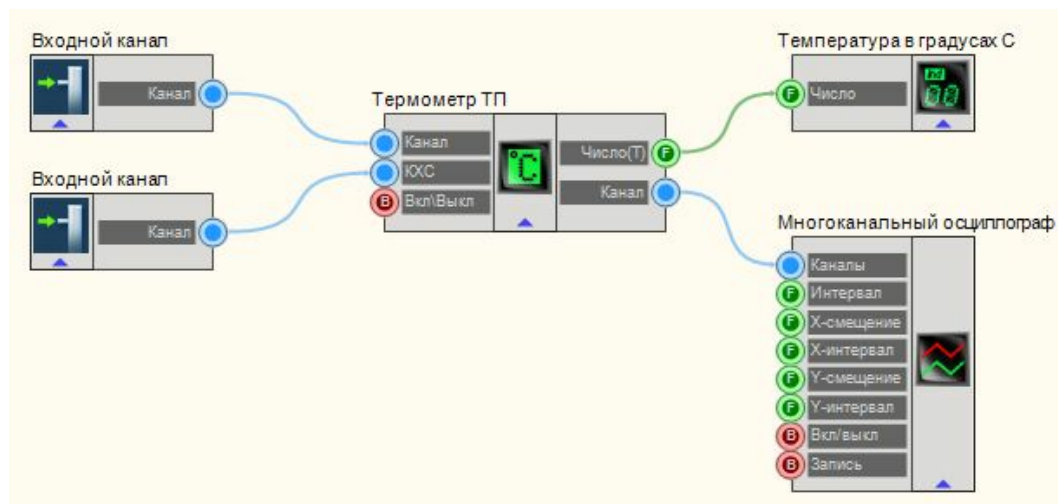
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- type (R-ТПП (13%родий/платина))- устанавливает тип подключенной к компоненту термопары:
  1. R-ТПП (13%родий/платина).
  2. R-ТПП (10%родий/платина).
  3. S-ТПР (платина/родий).
  4. J-ТЖК (железо/константан).
  5. T-ТМК (медь/константан).
  6. E-ТХКн (хромель/константан).
  7. K-ТХА (хромель/алюмель).
  8. N-ТНН (нихросил/нисил).
  9. A-ТВР (вольфрам/рений).
  10. L-ТХК (хромель/копель).
- Compensator (False) - задействование канала компенсатора холодного спая:
  1. False - выключение КХС.
  2. True - включение КХС.
- Activate (true) - состояние подачи сигнала (подан или отключен).



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Термометр термопар](#)<sup>[311]</sup> преобразует [Входной сигнал](#)<sup>[185]</sup> (Канал термопары) в канал с данными температуры. [Входной сигнал](#)<sup>[185]</sup> (Компенсатор холодного спая) служит для учета ЭДС, возникающей в контакте основной термопары с клеммами. [ЖК индикатор](#)<sup>[348]</sup> служит для вывода текущей температуры графически на интерфейс

оператора. [Многоканальный осциллограф](#)<sup>[494]</sup> нужен для просмотра изменений температуры во времени.

### Результат работы проекта



Данный пример непосредственно в ZETView

Более подробно об использовании Термометра термопар можно ознакомиться на официальном сайте, по ссылкам:

1. [Ссылка №1.](#)
2. [Ссылка №2.](#)
3. [Ссылка №3.](#)



### Математическое описание

Термопара — пара проводников из различных материалов, соединенных на одном конце и формирующих часть устройства, использующего термоэлектрический эффект для измерения температуры.

Принцип действия основан на эффекте Зеебека или, иначе, термоэлектрическом эффекте. Когда концы проводника находятся при разных температурах, между ними возникает разность потенциалов, пропорциональная разности температур. Коэффициент пропорциональности называют коэффициентом термоЭДС. У разных металлов коэффициент термоЭДС разный и, соответственно, разность потенциалов, возникающая между концами разных проводников, будет различной. Помещая спай из металлов с отличными коэффициентами термоЭДС в среду с температурой

T1, мы получим напряжение между противоположными контактами, находящимися при другой температуре T2, которое будет пропорционально разности температур T1 и T2.

Наиболее распространены два способа подключения термопары к измерительным преобразователям: простой и дифференциальный. В первом случае измерительный преобразователь подключается напрямую к двум термоэлектродам. Во втором случае используются два проводника с разными коэффициентами термоЭДС, спаянные в двух концах, а измерительный преобразователь включается в разрыв одного из проводников.

Для дистанционного подключения термопар используются удлинительные или компенсационные провода. Удлинительные провода изготавливаются из того же материала, что и термоэлектроды, но могут иметь другой диаметр. Компенсационные провода используются в основном с термопарами из благородных металлов и имеют состав, отличный от состава термоэлектродов.

Тип термопары	Буквенное обозначение НСХ*	Материал термоэлектродов		Коэффициент термоЭДС, мкВ/°С (в диапазоне температур, °С)	Диапазон рабочих температур, °С	Предельная температура при кратковременном применении, °С
		положительного	отрицательного			
ТЖК	J	Железо (Fe)	Сплав константен (45% Cu + 45% Ni, Mn, Fe)	50-64 (0-800)	от -200 до +750	900
ТХА	K	Сплав хромель (90,5% Ni + 9,5% Cr)	Сплав алюмель (94,5% Ni + 5,5% Al, Si, Mn, Co)	35-42 (0-1300)	от -200 до +1200	1300
ТМК	T	Медь (Cu)	Сплав константан (55% Си + 45% Ni, Mn,	40-60 (0-400)	от -200 до +350	400

			Fe)				
ТХК <sub>н</sub>	E	Сплав хромель (90,5% Ni + 9,5% Cr)	Сплав константан (55% Cu + 45% Ni, Mn, Fe)	59-81 (600)	(0- от-200 до+700		900
ТХК	L	Сплав хромель (90,5% Ni + 9,5% Cr)	Сплав копель (56% Си + 44% Ni}	64-88 (600)	(0- от -200 до +600		800
ТНН	N	Сплав никросил (83,49% Ni +13,7% Cr + 1,2% Si+ 0,15% Fe + 0,05% C + 0,01% Mg)	Сплав нисил (94,98% Ni + 0,02% Cr + 4,2% Si + 0,15% Fe + 0,05% C + 0,05% Mg)	26-36 (1300)	(0- от -270 до +1300		1300
ТПП1 3	R	Сплав платина-родий (87%Pt + 13%Rh)	платина (Pt)	10-14 (600-1600)	от 0 до +1300		1600
ТПП1 0	S	Сплав платина-родий (87% Pt — 13% Rh)	платина (Pt)	10-14 (600-1600)	от 0 до +1300		1600
ТПР	B	Сплав платина-родий (70% Pt - 30% Rh}	Сплав платина-родий (94% Pt-6%Rh)	10-14 (1000-1800)	от 600 до+1700		1800
ТВР	A-1 A-2 A-3	Сплав вольфрам-рений (95% W - 5% Re)	Сплав вольфрам-рений (80% W-20% Re)	14-7 (1300-2500)	от 0 до +2200 от 0 до +1800 от 0 до +1800		2500
ТСС	I	Сплав сильд	Сплав силин	-	от 0 до +800		900

В местах подключения проводников термопары к измерительной

системе возникают дополнительные термоЭДС. В результате их действия на вход измерительной системы фактически поступает сумма сигналов от рабочей термопары и от «термопар», возникших в местах подключения. Существуют различные способы избежать этого эффекта. Самым очевидным из них является поддержание температуры холодного спая постоянной.

На практике при измерении температур широко используется техника «компенсации холодного спая»: температура холодного спая измеряется другим датчиком температуры, а затем величина термоЭДС холодного спая программно или аппаратно вычитается из сигнала термопары. Места подключения термопары к измерительной системе должны иметь одинаковую температуру, то есть находиться в изотермальной зоне. Кроме того, в схеме с компенсацией холодного спая в этой же зоне должен находиться и датчик температуры холодного спая. Разработчик должен учитывать эти требования при конструировании измерительной системы.

Тип термопары	Особенности применения
ТХА	Обладают: — наиболее близкой к прямой характеристикой. Предназначены для работы в окислительных и инертных средах
ТХК	Обладают: — наибольшей чувствительностью; — высокой термоэлектрической стабильностью при температурах до 600°C. Предназначены для работы в окислительных и инертных средах. Недостаток: высокая чувствительность к деформациям
ТПП	Обладают: — хорошей устойчивостью к газовой коррозии, особенно на воздухе при высоких температурах; — высокой надежностью при работе в вакууме (но менее стабильны в нейтральных средах). Предназначены для длительной эксплуатации в окислительных средах. Недостаток: высокая чувствительность термоэлектродов к любым загрязнениям, появившимся при изготовлении, монтаже или эксплуатации термопар
ТВР	Обладают: — возможностью длительного применения при температурах до 2200 °C в неокислительных средах; — устойчивостью в аргоне, гелии, сухом водороде и азоте. Термопары с термоэлектродами из сплава платины с 10% родия относительно электрода из чистой платины могут

	использоваться как стандартные для установления номинальных статических характеристик термопар методом сравнения. Недостаток - плохая воспроизводимость термоЭДС, вынуждающая группировать термоэлектродные пары по группам с номинальными статическими характеристиками А-1, А-2, А-3
ТНН	Обладают: — высокой стабильностью термоЭДС (по сравнению с термопарами ТХА, ТПП, ТПР); — высокой радиационной стойкостью; — высокой стойкостью к окислению электродов. Предназначены в качестве универсального средства измерения температур в диапазоне температур 0-1230 °С

Термопары используются для измерения температуры различных типов объектов и сред, а также в автоматизированных системах управления и контроля. Термопары из вольфрам-рениевого сплава являются самыми высокотемпературными контактными датчиками температуры. Такие термопары незаменимы в металлургии для контроля температуры расплавленных металлов.

## 11.12.Термометр ТС

**Термометр термосопротивлений** - компонент. Предназначен для работы с данными, поступающими с канала термосопротивления. Служит для преобразования термоЭДС в численное значение температуры.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

➤ Входы:

- Канал 1 - канал, к которому подключается термосопротивление.
- Канал 2 - канал, к которому подключено нагрузочное сопротивление.
- Вкл\Выкл - параметр, отвечающий за включение и выключение термометра термопар.

➤ Выходы:

- Число(T) - вывод, с которого поступает усредненное за 0,1 сек значение температуры.
- Канал - виртуальный выходной канал, с которого поступают мгновенные значения температуры.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- type (Pt W = 1,391) - устанавливает тип подключенной к компоненту термосопротивления:
  1. Pt W = 1,391.
  2. Pt W = 1,385.
  3. Cu W = 1,428.
  4. Cu W = 1,426.
  5. Nc W = 1,617.
- Correction (0) - поправка в градусах.
- Activate (true) - состояние подачи сигнала (подан или отключен).



### **Пример**

Более подробно об использовании Термометр термосопротивлений можно ознакомиться на официальном сайте, по ссылкам:

1. [Ссылка №1.](#)
2. [Ссылка №2.](#)



### **Математическое описание**

Термометр сопротивления — датчик для измерения температуры, сопротивление чувствительного элемента которого зависит от температуры. Может быть выполнен из металлического или полупроводникового материала. В последнем случае называется термистором.



Действие термометров сопротивления основано на свойстве вещества изменять свое электрическое сопротивление с изменением температуры. При измерении температуры термометр сопротивления погружают в среду, температуру которой необходимо определить. Зная зависимость сопротивления термометра от температуры, можно по изменению сопротивления термометра судить о температуре среды, в которой он находится. При этом необходимо иметь в виду, что длина чувствительного элемента у большинства термометров сопротивления составляет несколько сантиметров, и поэтому при наличии температурных градиентов в среде термометром сопротивления измеряют некоторую среднюю температуру тех слоев среды, в которых находится его чувствительный элемент.

Температурный коэффициент электрического сопротивления обозначается  $\alpha$ . Этот коэффициент в общем виде может быть выражен равенством:

$$\alpha = \frac{1}{R} \frac{dR}{dt}$$

Температурный коэффициент электрического сопротивления принято определять от 0 до 100°C. Для этого случая выражение принимает вид:

$$\alpha_{0\ 100} = \frac{R_{100} - R_0}{100},$$

где  $R_0$  и  $R_{100}$  – сопротивления образца данного металла, измеренные соответственно при 0 и 100°C.

Относительное сопротивление  $W_t$  термометра определяется по формуле:

$$W_t = \frac{R_t}{R_0},$$

где  $R_t$  — сопротивление термометра при температуре  $t$ , Ом;  $R_0$  — сопротивление термометра при температуре 0°C (273,15 К), Ом.

Известно, что сплавы обладают меньшим значением температурного коэффициента сопротивления. Кроме того, воспроизводимость свойств сплавов далеко недостаточна по сравнению с чистыми металлами. Термометры сопротивления из чистых металлов, получившие наибольшее распространение, изготавливают обычно в виде обмотки из тонкой проволоки на специальном каркасе из изоляционного материала. Эту обмотку принято называть чувствительным элементом термометра сопротивления. В целях

предохранения от возможных механических повреждений и воздействия среды, температура которой измеряется термометром, чувствительный элемент его заключают в специальную защитную гильзу.

Наиболее распространённый тип термометров сопротивления — платиновые термометры. Это объясняется тем, что платина имеет высокий температурный коэффициент сопротивления и высокую стойкость к окислению. Эталонные термометры изготавливаются из платины высокой чистоты с температурным коэффициентом не менее 0,003925. В качестве рабочих средств измерений применяются также медные и никелевые термометры.

Для промышленных платиновых термометров сопротивления используется уравнение Каллендара-Ван Дьюзена, с известными коэффициентами, которые установлены экспериментально.

Для платины:

$$R_T = R_0[1 + AT + BT^2 + CT^3(T - 100)](-200^\circ\text{C} < T < 0^\circ\text{C}),$$

$$R_T = R_0[1 + AT + BT^2] (0^\circ\text{C} \leq T < 850^\circ\text{C}),$$

Здесь  $R_T$  сопротивление при  $T^\circ\text{C}$ ,  $R_0$  сопротивление при  $0^\circ\text{C}$ , и константы для платинового сопротивления:

$$A = 3,9083 \times 10^{-3} \text{ }^\circ\text{C}^{-1};$$

$$B = -5,775 \times 10^{-7} \text{ }^\circ\text{C}^{-2};$$

$$C = -4,183 \times 10^{-12} \text{ }^\circ\text{C}^{-4}.$$

Поскольку коэффициенты  $B$  и  $C$  относительно малы, сопротивление растёт почти линейно по мере роста температуры.

Для меди температурный коэффициент электрического сопротивления проводниковой меди лежит в пределах от  $4,2 \cdot 10^{-5}$  до  $4,27 \cdot 10^{-5} \text{ }^\circ\text{C}^{-1}$ .

Зависимость электрического сопротивления меди от температуры в широком интервале температур подчиняется уравнению

$$R_t = R_n(1 + \alpha t),$$

где  $R_t$  и  $R_0$  – сопротивления данного образца меди (чувствительного элемента медного термометра) соответственно при температуре  $t$  и  $0^\circ\text{C}$ ;  $\alpha$  – температурный коэффициент электрического сопротивления, характерный для данного образца медной проволоки, из которого изготовлен чувствительный термометра. Медная проволока, применяемая для изготовления чувствительных элементов медных термометров, имеет температурный коэффициент сопротивления  $\alpha = 4,26 \cdot 10^{-3} \text{ }^\circ\text{C}^{-1}$ .

Для никеля температурный коэффициент  $\alpha = 6,66 \cdot 10^{-3} \text{ }^\circ\text{C}^{-1}$ . Зависимость сопротивления от температуры имеет резко нелинейную характеристику.

Вследствие указанных выше причин термометры сопротивления чувствительным элементом из никелевой проволоки могут быть использованы для измерения температур не выше  $180\text{--}200^\circ\text{C}$ . Для изготовления термометров сопротивления для температур от  $-10$  до  $+180^\circ\text{C}$  может быть использована проволока из никеля. Ее электрическое сопротивление в интервале температур  $-10 \leq t \leq 180^\circ\text{C}$  может быть вычислено по формуле сопротивление в интервале температур  $-10 \leq t \leq 180^\circ\text{C}$  может быть вычислено по формуле

$$R_t = R_0(1 + At + Bt^2),$$

где  $R_t$ ,  $R_0$  — сопротивление чувствительного элемента термометра соответственно при температуре  $t$  и  $0^\circ\text{C}$ , Ом;

$$A = 5,86 \cdot 10^{-3} \text{ }^\circ\text{C}^{-1};$$

$$B = 8 \cdot 10^{-6} \text{ }^\circ\text{C}^{-2}.$$

Отношение сопротивлений  $R_{100}/R_0$  чувствительного элемента термометра из никеля должно находиться в пределах  $1,664 - 1,668$ . Максимальное отклонение значения сопротивления чувствительного элемента термометра при температуре  $t$  не должно превышать, Ом:

$$\Delta R = \pm(0,3 + 8,0 * 10^{-3})(R_{100} - R_0) * 100^{-1}$$

При этом отклонение сопротивления чувствительного элемента термометра при 0°C от его номинального значения (например,  $R_0 = 100 \text{ Ом}$ )

должно быть  $\pm 0,3(R_{100} - R_0) 100^{-1} \text{ Ом}$ .

Преимущества термометров сопротивления:

- Высокая точность измерений (обычно лучше  $\pm 1 \text{ }^\circ\text{C}$ ), может достигать до  $0,01 \text{ }^\circ\text{C}$ .
- Возможность исключения влияния изменения сопротивления линий связи на результат измерения при использовании 3-х или 4-х проводной схемы измерений
- Практически линейная характеристика

Недостатки термометров сопротивления

- Малый диапазон измерений (по сравнению с термопарами)
- Не могут измерять высокую температуру (по сравнению с термопарами).

### 11.13. Фазометр

**Фазометр** - компонент. Предназначен для определения разности фаз между двумя периодическими сигналами.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Канал 1 - входной канал № 1, на который поступает сигнал № 1 (базовый канал, относительно которого меряется фазовый сдвиг).
- Канал 2 - входной канал № 2, на который поступает сигнал № 2.
- Вкл\Выкл - параметр, отвечающий за включение и выключение фазометра.

➤ Выходные

- Фаза - выходной канал, с которого поступают численные значения разности фаз между двумя каналами (выводится фаза, относительно сигнала № 2).
- Канал - выходной виртуальный канал со значениями разности фаз двух входящих каналов.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

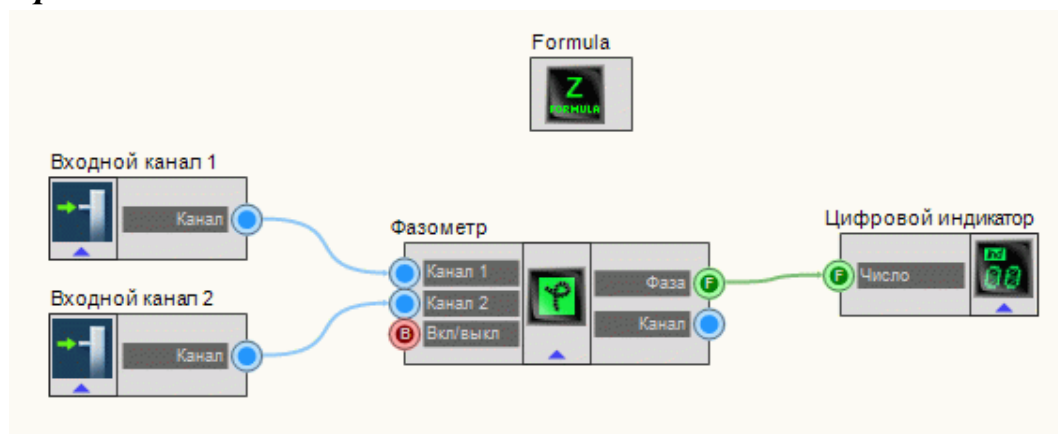
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- averagetime (0.1) - устанавливает время усреднения:
  1. 0.1 секунду.
  2. 1 секунду.
  3. 10 секунд.
- meashuretype (градусы) - тип измерения (градусы, радианы):
- Activate (true) - состояние подачи сигнала (подан или отключен).



## Пример

### Проект в SCADA ZETView



В этой схеме компонент **Фазометр** производит измерение разности фаз между каналами [Входной канал 1](#)<sup>[185]</sup> и [Входной канал 2](#)<sup>[185]</sup>. [ZETFormula](#)<sup>[322]</sup> служит для генерирования двух синусоидальных сигналов, причем одна из синусоид имеет фазовый сдвиг относительно другой. Компонент [Цифровой](#)

[индикатор](#)<sup>[362]</sup> служит для графического вывода значения разности фаз.

### *Результат работы проекта*



### *Математическое описание*

Фазометр — электроизмерительный прибор, предназначенный для измерения углов сдвига фаз между двумя изменяющимися периодически электрическими колебаниями, например в трёхфазной системе электроснабжения

Под активной мощности  $P$  понимают среднее значение мгновенной мощности  $p$  за период  $T$ :

$$p = \frac{1}{T} \int_0^T p \, dt = \frac{1}{T} \int_0^T ui \, dt$$

Если ток:

$$i = I_m \sin \omega t$$

А напряжение на участке цепи

$$u = U_m \sin(\omega t + \varphi)$$

$$p = \frac{1}{T} \int_0^T I_m U_m \sin \omega t \sin(\omega t + \varphi) \, dt = \frac{U_m I_m}{2} \cos \varphi$$

Выражение для эффективной мощности в цепи синусоидального тока имеет вид:

$$P = V * I * \cos \varphi$$

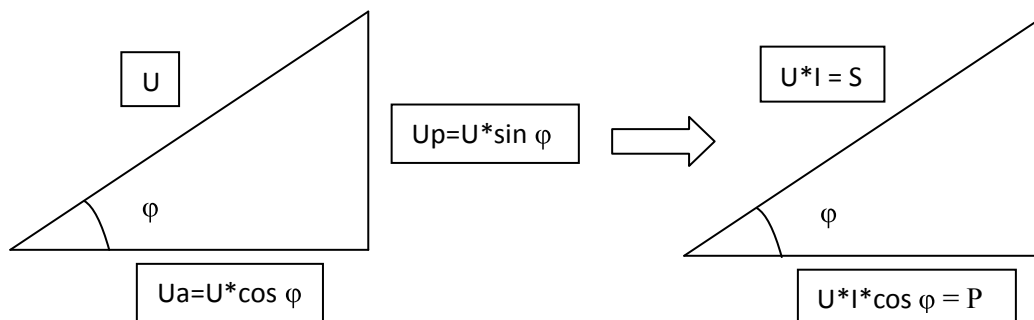
где  $V$  и  $I$  – среднеквадратичные значения напряжения и тока. Входящие в выражение множитель  $\cos \varphi$  называется коэффициентом мощности. Единицей измерения эффективной мощности является [Вт].

- В чисто резистивных цепях ( $\varphi = 0$ ) коэффициент мощности равен 1, а эффективная мощность определяется выражением  $P=V*I$ .
- В чисто реактивных цепях ( $\varphi = \pm 90^\circ$ ) коэффициент мощности, а, следовательно, и эффективная мощность равны 0.
- В резистивно-емкостных и резистивно-индуктивных цепях ( $-90^\circ < \varphi < 90^\circ$ ) эффективная мощность всегда положительная.

Выражение для реактивной мощности в цепи синусоидального тока имеет вид:

$$Q = U * I * \sin \varphi$$

Из треугольника напряжений:



Выражение для полной мощности в цепи синусоидального тока имеет вид:

$$S = U * I$$


Эффективная электрическая мощность может быть преобразована в другие формы мощности (тепловую, механическую и т. д.). А реактивная мощность, в свою очередь, не может быть преобразована в любые другие виды энергий.

Фазометр позволяет определить и знак угла, на основании чего можно сделать вывод и о типе нагрузке (резистивно-емкостная, резистивно-индуктивная и т. д.).

## 11.14.Частотомер

**Частотомер** - компонент. Предназначен для определения частоты периодического процесса или частот гармонических составляющих спектра сигнала.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - входной канал, на который поступает сигнал, у которого нужно измерить требуемые величины.
- Вкл\Выкл - параметр, отвечающий за включение и выключение вольтметра.

#### ➤ Выходные

- Частота - выходной канал, с которого в потоке поступают уже измеренные значения частоты сигнала, Гц

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

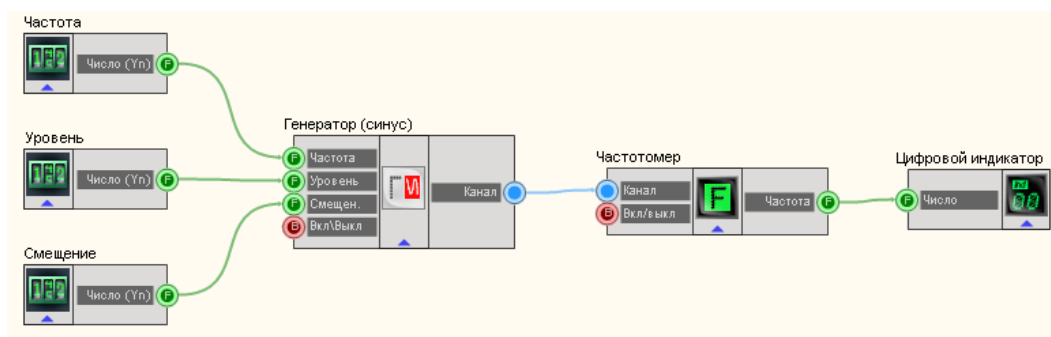
- averagetime (0.1) - устанавливает время усреднения:
  1. 0.1 секунду.
  2. 1 секунду.
  3. 10 секунд.
- Activate (true) - состояние подачи сигнала (подан или отключен).



### *Пример*

### *Проект в SCADA ZETView*





В этой схеме компонент "Частотомер" измеряет частоту синусоидального сигнала, поступающего с [Синусоидальный сигнал](#)<sup>[247]</sup>. [Селекторы](#)<sup>[213]</sup> служат для установки значений частоты, уровня, смещения синусоидального сигнала. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления значения частоты сигнала.

### Результат работы проекта

Частота, Гц

00384.00

Смещение, В

00001.00

Уровень, В

00001.00

Частотомер, мВ

51.0

Данный пример непосредственно в ZETView

## 11.15.Энкодер

**Энкодер** - компонент. Компонент предназначен для измерения относительного положения (перемещения), скорости и направления перемещения при помощи оптических датчиков угловых или линейных перемещений (энкодеров), подключенных к входным каналам АЦП. Программа Энкодер порождает виртуальные каналы перемещения и скорости перемещения.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал А - контакт, к которому подключается фаза (канал) А.
- Канал Б - контакт, к которому подключается фаза (канал) Б.
- Канал Б - включение канала фазы Б.
- Метка - контакт, к которому подключается канал метки.
- Метка - параметр, отвечающий за включение и выключение термометра термопар.
- Сброс - сброс всех значений компонентов ноль.

#### ➤ Выходные

- Канал х - канал перемещения.
- Канал V - канал скорости.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- channelA\_name - имя канала измерений фазы А.
- channelB\_name - имя канала измерений фазы Б.
- channelLabel\_name - имя канала измерений метки.
- channelB\_enable (true) - включение/выключение канала фазы Б.
- channelLabel\_enable (true) - включение/выключение канала метки.
- channelPath\_enable (true) - включение/выключение виртуального канала перемещений.
- channelSpeed\_enable (true) - включение/выключение виртуального канала скорости.
- units (мм) - выбор единицы измерения:
  - мм - миллиметр.
  - см - сантиметр.
  - м - метр.
  - градус - градус.
  - оборот - оборот.

- resolution (250) - разрешающая способность энкодера (задается в метках на единицу измерения).
- Level\_autoSet (true) - включение/выключение автоматической установки порога.
- Level\_low (-2000.000000)- верхний порог (в единицах измерения канала).
- Level\_high (2000.000000)- нижний порог (в единицах измерения канала).
- Inverction (false) - инверсия направления перемещения (необходимое условие - включенная фаза Б)



### Пример

Более подробно об использовании Энкодера :

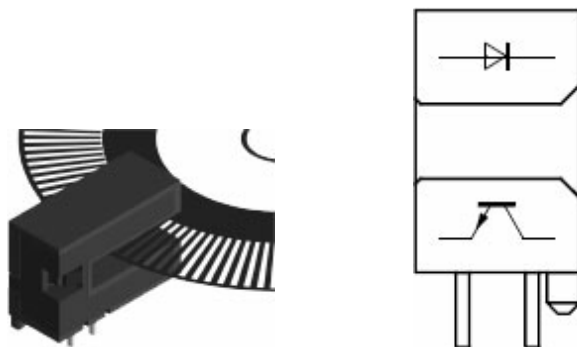
#### Энкодер

Программа "Энкодер" предназначена для измерения относительного положения (перемещения), скорости и направления перемещения при помощи оптических датчиков перемещения (энкодеров), подключенных к входным каналам модулей АЦП и анализаторов спектра.

На базе оптических датчиков создаются датчики линейных и угловых перемещений. Точности таких датчиков могут быть от 1 мкм до 1 мм при длине измерительной базы от 8 мм до 3 м. Датчики угловых перемещений могут иметь от 100 до 10000 маркеров на один оборот, т.е. разрешение может быть до 5 минут.

Оптическая технология предложила ряд классических способов для построения энкодера - датчика, представляющего информацию о движении, положении или направлении либо непосредственно в цифровой форме, либо генерирующего последовательность импульсов, из которой после оцифровки может быть сформирован цифровой код.





Принцип работы энкодеров проиллюстрирован на рисунке 1. Оптический энкодер состоит из тонкого оптического диска и стационарного блока - измерительной головки, включающей в себя источник света и фотодетектор. Оптический диск включает поверхность из прозрачных и непрозрачных участков. Маркерами могут быть, например, отверстия в металлическом листе или метки на стеклянном диске. При вращении диска, в зависимости от его типа, маркеры пропускают или перекрывают луч света, направленный от светового источника к фотоприемнику.

Фотодетектор генерирует сигнал частотой, равной частоте следования кодовых элементов, в цифровой форме или аналоговый импульсный сигнал, который также может быть усилен и оцифрован. При добавлении второй пары "светодиод-фототранзистор" с угловым смещением относительно первой, соответствующим четверти периода сигнала, может быть получена вторая последовательность импульсов - канал Б с фазовым смещением относительно канала А на  $90^\circ$ . Инкрементальный энкодер, который использует три оптических датчика, позволяет одновременно удваивать разрешение при измерении положения и скорости и детектировать направление.

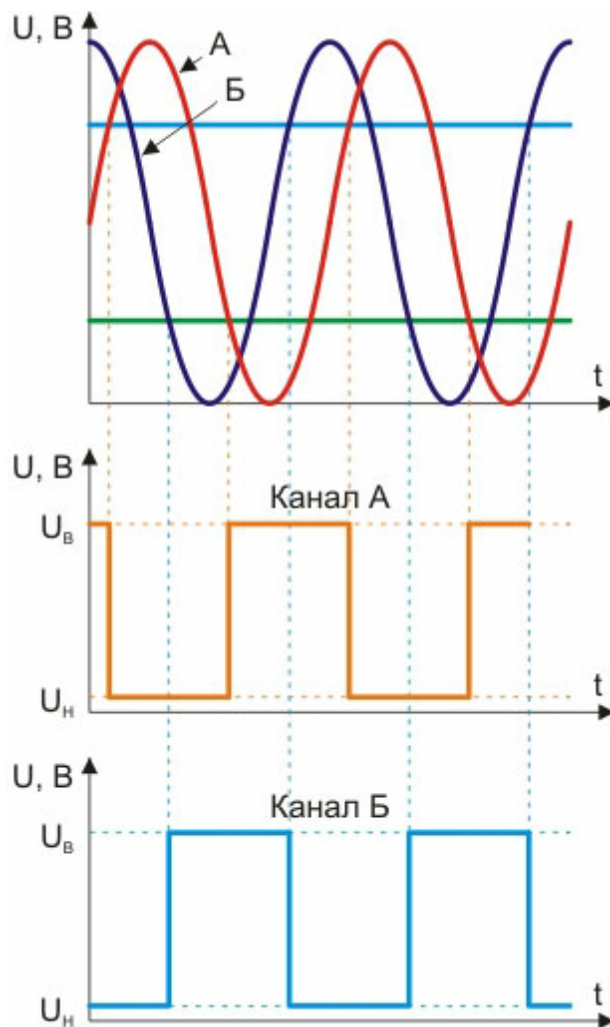


Рисунок 1

Датчики линейного и углового перемещения подключаются к модулям АЦП напрямую. Для питания датчиков можно использовать выход генератора. Разрешение инкрементальных энкодеров измеряется в импульсах за оборот (pulses per revolution, ppr). В программе "Энкодер" пользователю предоставляется возможность выбора разрешения используемого энкодера (окно "Разрешение, меток/е.и."). "Е.и." - единица измерения, которая может быть выбрана из ряда "мм, см, м, гр. (градусы), об. (обороты)" или прописано вручную в окне "Единица измерения".

Также в ниспадающих списках "Фаза А" и "Фаза Б" выбираются каналы подключения энкодера, к которым подключены соответствующие сигналы "Канал А" и "Канал Б". Ниспадающий список "Метка 0" предназначена для выбора канала модуля АЦП или анализатора спектра, к которому подключен сигнал синхронизации. Пороги срабатывания сигнала синхронизации устанавливаются вручную или автоматически.

При установленных галочках "Перемещение" и "Скорость" в окне программы "Энкодер" в сервере данных создаются дополнительные виртуальные каналы, содержащие соответственно информацию о перемещении и скорости.

На рисунке 2 приведены формы сигналов с каналов "Фаза А", "Фаза Б" и "Метка 0", полученные при помощи программы "Многоканальный осциллограф". На рисунке 3 показаны формы сигналов с виртуальных каналов перемещения и скорости и сигнал "Метка 0".



Рисунок 2

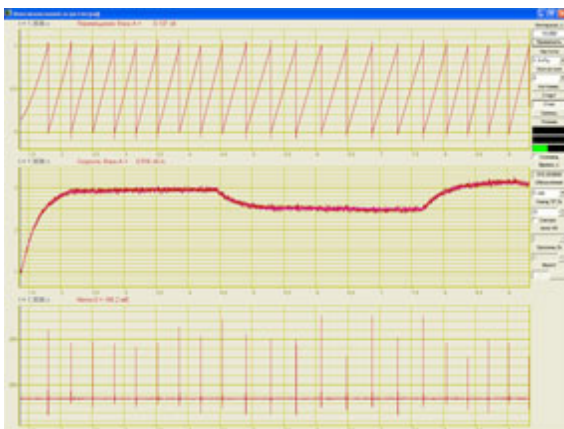


Рисунок 3

При использовании программы "Энкодер" совместно с программами из состава ZETLab и датчиками угловых перемещений, пользователь может проводить анализ крутильных колебаний и использовать эту аппаратуру для замены торсиографов.

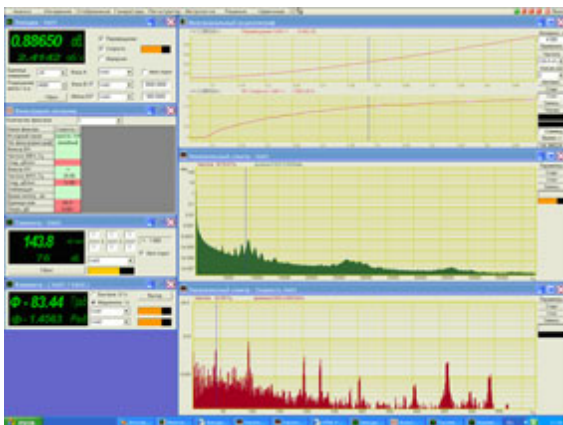


Рисунок 4

Встроенный в программу модуль управления и автоматизации из состава ZETLab Studio обеспечивает простоту и удобство при построении собственных программно-измерительных комплексов.

#### Назначение и описание программы Энкодер

##### *Назначение программы*

Программа Энкодер предназначена для измерения относительного положения (перемещения), скорости и направления перемещения при помощи оптических датчиков угловых или линейных перемещений (энкодеров), подключенных к входным каналам АЦП. Программа Энкодер порождает виртуальные каналы перемещения и скорости перемещения. Эти каналы доступны для последующего анализа другими программами из состава ZETLab.

На базе оптических датчиков создаются датчики линейных и угловых перемещений. Точность таких датчиков может быть от 1 мкм до 1 мм при длине измерительной базы от 8 мм до 3 м. Датчики угловых перемещений могут иметь от 100 до 10000 маркеров на один оборот, т.е. разрешение может быть от нескольких градусов до 5 минут.

Оптическая технология предложила ряд классических способов для построения энкодера-датчика представляющего информацию о движении, положении или направлении либо непосредственно в цифровой форме (абсолютные энкодеры), либо генерирующего последовательность импульсов (инкрементальные энкодеры).

Далее в описании программы Энкодер речь пойдет только об инкрементальных энкодерах, так как программа Энкодер построена для работы только с таким типом этих датчиков.

Оптический энкодер состоит из тонкого оптического диска и стационарного блока измерительной головки, включающей в себя источник света и фотодетектор. Оптический диск содержит поверхность из прозрачных и непрозрачных участков. Маркерами могут быть, например, отверстия в металлическом листе или метки на стеклянном диске. При вращении диска, в зависимости от его типа, маркеры пропускают или перекрывают луч света, направленный от светового источника к фотоприемнику.

Фотодетектор генерирует сигнал с частотой, равной частоте следования кодовых элементов в цифровой форме или аналоговый импульсный сигнал, который также может быть усилен и оцифрован. При добавлении второй пары светодиодфототранзистор с угловым смещением относительно первой, соответствующим четверти периода сигнала, может быть получена вторая последовательность импульсов - канал Б с фазовым смещением относительно канала А на  $90^\circ$ . Инкрементальный энкодер, который использует два оптических канала, позволяет одновременно удваивать разрешение при измерении положения и скорости и определять направление. Третий канал используется для привязки к начальной отметке (метка «0»).

#### *Описание программы*

Для запуска программы Энкодер необходимо из меню Измерение панели ZETLab выбрать команду Энкодер. На экране монитора отобразится рабочее окно программы Энкодер. В заголовке окна программы будет отображаться название программы и название канала, к которому подключен канал А (фаза А) оптического датчика перемещения (энкодера).

Примечание: программу Энкодер можно запустить непосредственно из рабочей директории ZETLab (по умолчанию: c:\ZETLab\). Имя запускаемого файла: Encoder.exe












Измерение	Отображение	Ге
 Вольтметр переменного тока		
 Вольтметр постоянного тока		
 Селективный вольтметр		
 Частотомер		
 Фазометр		
 Тахометр		
 Энкодер		
 Омметр		
 Термометр ТС		
 Термометр ТП		
 Тензомер		



Рис.1

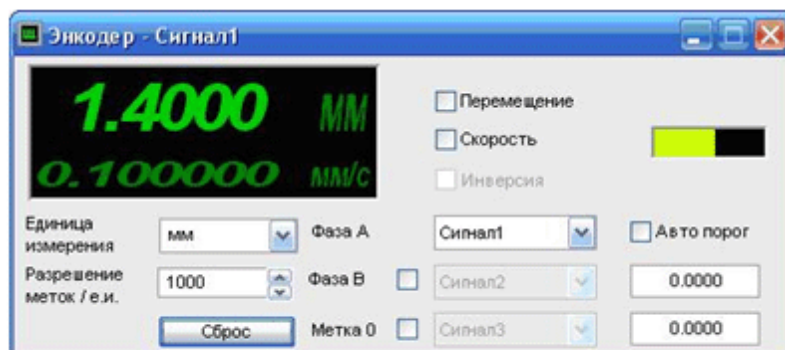


Рис.2

В левой верхней части рабочего окна программы Энкодер расположен графический индикатор, в котором отображаются измеренные угловое или линейное перемещение (положение) и скорость перемещения, информация о которых поступает с выбранного (выбранных) канала. В зависимости от поставленной задачи перемещение измеряется в указанных единицах измерения, а скорость перемещения в единицах измерения в секунду. Например, для измерения перемещение в миллиметрах необходимо в списке Единица измерения (список описывается ниже) указать единицу измерения миллиметры (мм), после этого перемещение на графическом индикаторе будет отображаться в мм, а скорость перемещения в мм/с (миллиметров в секунду).

В списке, расположенном справа от надписи Единица измерения, выбирается либо вводится с клавиатуры единица измерения, в которой будет отображаться сигнал по выбранному каналу. Для выбора необходимой единицы измерения необходимо нажать левой кнопкой «мыши» на кнопку списка, и, в раскрывшемся списке, выбрать нужную единицу измерения. Если в списке нет необходимой единицы измерения, то, нажав правой клавишей «мыши» на поле списка, с клавиатуры ввести требуемую единицу измерения.

Разрешение инкрементальных энкодеров определяется количеством импульсов за один оборот (pulses per revolution, ppr). В списке, расположенном справа от надписи Разрешение меток/е.и., выбирается необходимое количество меток в установленную единицу измерения. Например, инкрементальный энкодер углового перемещения имеет 1080 меток на один оборот, соответственно 3 метки на один градус поворота. Необходимо

измерять положение энкодера в градусах с точность измерения в один градус. Для этого, в списке Единица измерения выбирается единица измерения – градусы (гр.), а в списке Разрешение меток/е.и. устанавливается 3 (три метки на один градус поворота энкодера). Устанавливается необходимое разрешение нажатием левой кнопки «мышь» по кнопкам списка разрешения, либо, нажав левой кнопкой «мышь» по полю списка разрешения, прокрутив ролик «мышь», либо, нажав левой кнопкой «мышь» по полю списка разрешения, ввести значение с клавиатуры и нажать клавишу.

Под полем списка Разрешение меток/е.и. располагается кнопка Сброс, нажатие которой приводит к обнулению значения перемещения (положения).

Флажки Перемещение и Скорость, расположенные справа от графического индикатора, разрешают включение/выключение виртуальных каналов Перемещение и Скорость, порождаемых программой Энкодер. Эти каналы доступны для последующего анализа другими программами. Установленные флажки – виртуальные каналы включены, снятые – выключены. Данные в этих виртуальных каналах идут в темпе обработки каждого импульса без усреднения. Это позволяет исследовать не только перемещение и скорость перемещения, но и их неравномерность. При включении энкодера совместно с другими датчиками, например, давления или температуры, данные с этих датчиков и данные перемещения и скорости перемещения идут синхронно с точностью до одного импульса энкодера.

Флажок Инверсия, который становится доступным для установки или снятия при установленном флажке Фаза В, позволяет инвертировать сигнал направления перемещения. Установленный флажок – происходит инверсия, снятый – сигнал не инвертирован.

Для измерения перемещения и скорости перемещения по физическому каналу, к которому подключен канал А энкодера, необходимо в поле списка (со стрелкой), расположенному справа от надписи Фаза А, выбрать название этого канала.

При использовании в измерениях канала В энкодера необходимо установить флажок, расположенный справа от надписи Фаза В, и, в ставшем доступным поле списка выбрать название включенного физического канала, к которому подключен канал В энкодера. Снятие флажка Фаза В блокирует поле списка выбора канала В и флажок установки инверсии.

При необходимости вычисления абсолютного положения необходимо установить флажок, расположенный справа от надписи Метка 0, и, в ставшем доступным поле списка выбрать название включенного физического

канала, к которому подключен канал нулевой метки энкодера. Каждый раз, при прохождении нулевой метки пары светодиодфототранзистор происходит обнуление на графическом индикаторе измеренных показаний перемещения. Например, это удобно при измерении линейных перемещений, при которых происходит возвратнопоступательное перемещение. Установив нулевую метку посередине можно измерять перемещение в ту или другую сторону относительно установленной нулевой метки.

Флажок Авто порог служит для включения/выключения автоматической/ручной установки верхнего и нижнего порога входного уровня, по которому будут производиться измерения перемещения. Установленный флажок – программа сама автоматически устанавливает верхний и нижний порог уровня сигнала. Снятый флажок – разрешен ручной ввод верхнего и нижнего порога уровня сигнала.

Поля ввода, расположенные под флажком Авто порог, служат для установки верхнего и нижнего порога в ручном режиме (флажок Авто порог снят). Верхнее поле ввода служит для установки верхнего порога, нижнее – для установки нижнего порога. Ввод значений верхнего и нижнего порога осуществляется с клавиатуру. После ввода значений нажать клавишу клавиатуры. При установленном флажке Авто порог поля ввода верхнего и нижнего порога не доступны для ввода значений. Верхний и нижний порог устанавливается для исключения ложных срабатываний при измерении перемещения. Для правильного измерения перемещения верхний порог не должен превышать максимального уровня сигнала по этому каналу, нижний порог не должен быть ниже минимального уровня. Определить максимальный и минимальный уровень сигнала по каналу, к которому подключен энкодер, можно, например, запустив программу Многоканальный осциллограф выбрать этот канал и по осциллограмме оценить эти уровни. Для выхода из программы необходимо нажать кнопку, расположенную в правом верхнем углу окна.

При использовании программы Энкодер совместно с программами из состава ZETLab и датчиками угловых перемещений, пользователь может проводить анализ крутильных колебаний и использовать эту аппаратуру для замены торсиографов.

#### Подключение датчиков

Датчики линейных или угловых перемещений подключаются к входным каналам (АЦП) устройств, производимых ЗАО «Электронные технологии и метрологические системы», напрямую.

После подключения энкодера к входным каналам необходимо в программе Настройка параметров АЦП и ЦАП включить эти каналы либо убедиться, что они включены.

Для измерения перемещения (положения) и скорости перемещения необходимо в программе Редактирование файлов параметров настроить параметры каналов, к которым подключен энкодер. Параметры измерительных каналов должны быть настроены относительно измерения напряжения.

Каналы для измерения выбираются любые, название каналов пользователь вводит по своему усмотрению.

Для питания датчиков можно использовать как выход встроенного генератора (при условии комплектации встроенным генератором) в режиме генерирования синусоидального сигнала с постоянным смещением нуля, так и внешний источник питания.



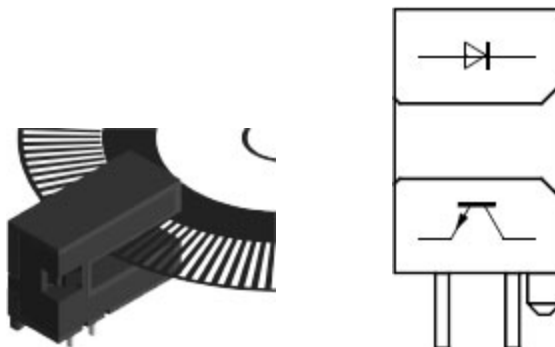
### *Математическое описание*

**Энкодер** — устройство, предназначенное для преобразования угла поворота вращающегося объекта (вала) в электрические сигналы, позволяющие определить угол его поворота. Энкодеры подразделяются на инкрементальные и абсолютные. Инкрементальный энкодер выдает за один оборот определенное количество импульсов. А абсолютные энкодеры позволяют в любой момент времени знать текущий угол поворота оси, в том числе и после пропадания и восстановления питания. А многооборотные абсолютные энкодеры, кроме того, также подсчитывают и запоминают количество полных оборотов оси. Энкодеры могут быть как оптические, резисторные, так и магнитные и могут работать через шинные интерфейсы или промышленную сеть.

Инкрементальные энкодеры предназначены для определения угла поворота вращающихся объектов. Они генерируют последовательный

импульсный цифровой код, содержащий информацию относительно угла поворота объекта. Если вал останавливается, то останавливается и передача импульсов. Основным рабочим параметром датчика является количество импульсов за один оборот. Мгновенную величину угла поворота объекта определяют посредством подсчёта импульсов от старта. Для вычисления угловой скорости объекта процессор в тахометре выполняет дифференцирование количества импульсов во времени, таким образом, показывая сразу величину скорости, то есть число оборотов в минуту. Выходной сигнал имеет два канала, в которых идентичные последовательности импульсов сдвинуты на  $90^\circ$  относительно друг друга (парафазные импульсы), что позволяет определять направление вращения. Имеется также цифровой выход нулевой метки, который позволяет всегда рассчитать абсолютное положение вала.

Принцип работы инкрементальных энкодеров проиллюстрирован на рисунке 1. Оптический энкодер состоит из тонкого оптического диска и стационарного блока - измерительной головки, включающей в себя источник света и фотодетектор. Оптический диск содержит поверхность из прозрачных и непрозрачных участков. Маркерами могут быть, например, отверстия в металлическом листе или метки на стеклянном диске. При вращении диска, в зависимости от его типа, маркеры пропускают или перекрывают луч света, направленный от светового источника к фотоприемнику.



Фотодетектор генерирует цифровой сигнал с частотой, равной частоте следования кодовых элементов или аналоговый импульсный сигнал, который также может быть усилен и оцифрован. При добавлении второй пары светодиод-фототранзистор с угловым смещением относительно первой, соответствующим четверти периода сигнала, может быть получена вторая последовательность импульсов - канал Б с фазовым смещением относительно канала А на  $90^\circ$ . Инкрементальный энкодер, который использует два оптических канала, позволяет одновременно удваивать разрешение при измерении положения и скорости и определять направление. Третий канал используется для привязки к начальной отметке (метка «0»).

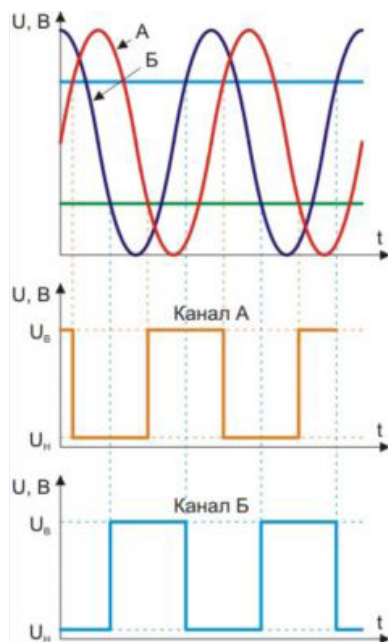


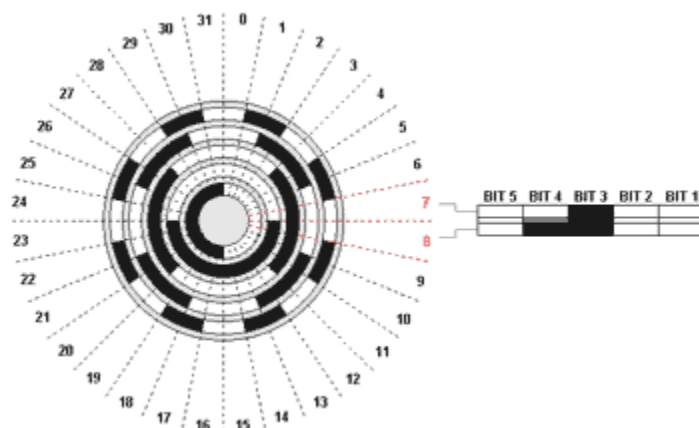
Рис.1

Абсолютные энкодеры, как оптические, так и магнитные имеют своей основной рабочей характеристикой число шагов, то есть уникальных кодов на оборот и количество таких оборотов, при этом не требуется первичной установки и инициализации датчика. Поэтому абсолютные энкодеры не теряют свою позицию при исчезновении напряжения.

Наиболее распространённые типы выходов сигнала — это код Грея, параллельный код, интерфейсы Profibus-DP, CANopen, DeviceNet, SSI, LWL, через которые также осуществляется программирование датчиков.

Абсолютный энкодер относится к типу энкодеров, который выполняет уникальный код для каждой позиции вала. В отличие от инкрементного энкодера, счетчик импульсов не нужен, т.к. угол поворота всегда известен. Абсолютный энкодер формирует сигнал как во время вращения, так и в режиме покоя. Диск абсолютного энкодера отличается от

диска пошагового энкодера, так как имеет несколько концентрических дорожек. Каждой дорожкой формируется уникальный двоичный код для конкретной позиции вала.



Абсолютный энкодер не теряет своего значения при потере питания и не требует возвращения в начальную позицию. Сигнал абсолютного энкодера не подвержен помехам и для него не требуется точная установка вала. Кроме того, даже если кодированный сигнал не может быть прочитан энкодером если, например, вал вращается слишком быстро, правильный угол вращения будет зарегистрирован, когда скорость вращения уменьшится. Абсолютный энкодер устойчив к вибрациям.

Код Грея предпочтительнее обычного двоичного тем, что обладает свойством непрерывности бинарной комбинации: изменение кодируемого числа на единицу соответствует изменению кодовой комбинации только в одном разряде. Он строится на базе двоичного по следующему правилу: старший разряд остается без изменения; каждый последующий разряд инвертируется, если предыдущий разряд исходного двоичного кода равен единице. Этот алгоритм построения может быть формально представлен как результат сложения по модулю два исходной комбинации двоичного кода с



такой же комбинацией, но сдвинутой на один разряд вправо. При этом крайний правый разряд сдвинутой комбинации отбрасывается.

Таким образом, Грей-код является так называемым одношаговым кодом, т.к. при переходе от одного числа к другому всегда меняется лишь какой-то один бит. Погрешность при считывании информации с механического кодового диска при переходе от одного числа к другому приведет лишь к тому, что переход от одного положения к другому будет лишь несколько смещен по времени, однако выдача совершенно неверного значения углового положения при переходе от одного положения к другому полностью исключается. Преимуществом Грей-кода является также его способность зеркального отображения информации. Так, инвертируя старший бит можно простым образом менять направление счета и, таким образом, подбирать к фактическому (физическому) направлению вращения оси. Изменение направления счета может легко изменяться, управляя так называемым входом Complement. Выдаваемое значение может быть возрастающим или спадающим при одном и том же физическом направлении вращения оси.

Поскольку информация, выраженная в Грей-коде, имеет чисто кодированный характер не несущей реальной числовой информации, должен он перед дальнейшей обработкой сперва преобразован в стандартный бинарный код. Осуществляется это при помощи преобразователя кода (декодера Грей-Бинар), который к счастью легко реализуется с помощью цепи из логических элементов «исключающее или» ( XOR ) как программным, так и аппаратным способом (см. схему ниже).

Из таблицы видно, что при переходе от одного числа к другому

(соседнему) лишь один бит информации меняет свое состояние, если число представлено кодом Грея, в то время, как в двоичном коде могут поменять свое состояние несколько бит одновременно. Код Грея - выход, следовательно, он никогда не имеет ошибку чтения и применяется во многих абсолютных энкодерах.

Десятичный код	Двоичный код				Код Грея
	$2^3$	$2^2$	$2^1$	$2^0$	
0	0	0	0	0	0 0 0 0
1	0	0	0	1	0 0 0 1
2	0	0	1	0	0 0 1 1
3	0	0	1	1	0 0 1 0
4	0	1	0	0	0 1 1 0
5	0	1	0	1	0 1 1 1
6	0	1	1	0	0 1 0 1
7	0	1	1	1	0 1 0 0
8	1	0	0	0	1 1 0 0
9	1	0	0	1	1 1 0 1
10	1	0	1	0	1 1 1 1
11	1	0	1	1	1 1 1 0
12	1	1	0	0	1 0 1 0
13	1	1	0	1	1 0 1 1
14	1	1	1	0	1 0 0 1
15	1	1	1	1	1 0 0 0

Биты меняющие свое состояние, при переходе от одного числа к другому, обозначены красным цветом.

### Однооборотный энкодер.

Однооборотными ( Single - Turn ) датчиками называются датчики, которые выдают абсолютные значения в пределах одного оборота, т.е. в радиусе 360°. После одного оборота код является полностью пройденным и

начинается опять с его начального значения. Эти датчики служат, преимущественно, для измерения угла поворота и применяются, например, в антенных системах, эксцентричных коленчатых прессах и т.д.

### **Многооборотный энкодер**

Линейные перемещения предполагают необходимым применение измерительной системы с  $n$  - количеством оборотов. Например, при линейных приводах или при задачах измерения с помощью зубчатой измерительной штанги, применение однооборотных датчиков является неприемлемым. В этом случае приходят на помощь датчики, где дополнительно к измерению угла поворота в пределах одного оборота также происходит регистрация количества оборотов с помощью дополнительно встроенного передаточного механизма, т.е. своего рода редуктора из нескольких кодовых оптических дисков, образуя, таким образом, многооборотный энкодер ( Multi - Turn ).

### **Оптические энкодеры**

Оптические энкодеры имеют жёстко и закреплённый соосно валу стеклянный диск с прецизионной оптической шкалой. При вращении объекта оптопара считывает информацию, а электроника преобразовывает её в последовательность дискретных электрических импульсов. Абсолютные оптические энкодеры — это датчики угла поворота, где каждому положению вала соответствует уникальный цифровой выходной код, который наряду с числом оборотов является основным рабочим параметром датчика. Абсолютные оптические энкодеры, так же как и инкрементальные энкодеры, считывают и фиксируют параметры вращения оптического диска.

### **Магнитные энкодеры**

Магнитные энкодеры с высокой точностью регистрируют прохождение магнитных полюсов вращающегося магнитного элемента непосредственно вблизи чувствительного элемента, преобразуя эти данные в соответствующий цифровой код.

### **Механические и оптические энкодеры с последовательным выходом**

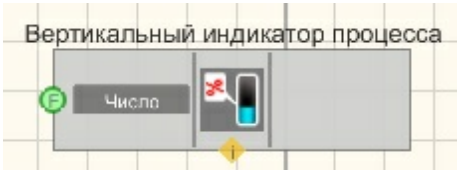
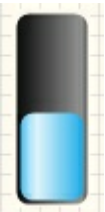
Содержат диск из диэлектрика или стекла с нанесёнными выпуклыми, проводящими или непрозрачными участками. Считывание абсолютного угла поворота диска производится линейкой переключателей или контактов в случае механической схемы и линейкой оптронов в случае оптической. Выходные сигналы представляют собой код Грея, позволяющий избавиться от неоднозначности интерпретации сигнала.

## Глава 12.Индикация

### 12.1.Вертикальный индикатор процесса

**Вертикальный индикатор процесса** - компонент. Предназначен для графического отображения состояния протекания процесса с разными настройками.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

**Параметры:**

## ➤ Входные:

- Число - вход, на который поступает число, которое будет отображаться на индикаторе процесса.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

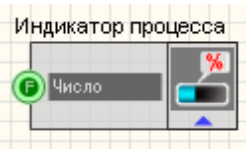

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- maxValue (0) - максимальное значение, которое сможет отобразить индикатор.
- curValue (0) - позволяет установить текущее состояние индикатора.
- fillColorUp (00a2e8) - установка верхнего цвета градиента индикатора.
- fillColorDown (ffffff) - установка нижнего цвета градиента индикатора.
- backColorUp (000000) - установка верхнего цвета градиента фона.
- backColorDown (c0c0c0) - установка нижнего цвета градиента фона.
- FrameVisible (false) - установка видимости рамки.
- ProcentVisible (false) - установка отображения процентов.
- style (выпуклый) - установка стиль процессбара:
  - 1 - косые линии
  - 2 - раздробленный
  - 3 - выпуклый
  - 4 - Виста
- AnimationEnable (true) - включение/отключение анимации.
- Enable (true) - Включение/отключение компонента.

## 12.2. Горизонтальный индикатор процесса

**Индикатор процесса** - компонент. Предназначен для графического отображения состояния протекания процесса с разными настройками.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

**Параметры:**

## ➤ Входные:

- Число - вход, на который поступает число, которое будет отображаться на индикаторе процесса.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- maxValue (0) - максимальное значение, которое сможет отобразить индикатор.
- currValue (0) - позволяет установить текущее состояние индикатора.
- fillColorUp (00a2e8) - установка верхнего цвета градиента индикатора.
- fillColorDown (ffffff) - установка нижнего цвета градиента индикатора.
- backColorUp (000000) - установка верхнего цвета градиента фона.
- backColorDown (c0c0c0) - установка нижнего цвета градиента фона.
- FrameVisible (false) - установка видимости рамки.
- ProcentVisible (false) - установка отображения процентов.
- style (выпуклый) - установка стиль процессбара:
  - 1 - косые линии
  - 2 - раздробленный
  - 3 - выпуклый
  - 4 - Виста
  - 5 - Объемный.
- AnimationEnable (true) - включение/отключение анимации.
- Enable (true) - Включение/отключение компонента.

## 12.3. ЖК индикатор

**ЖК индикатор** - компонент. Предназначен для графического отображения численных параметров и значений.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

**Параметры:**

## ➤ Входящие:

- Число - вывод, к которому подключается канал с численным значением какого-либо параметра.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IndValue (0) - отображаемое значение, которое поступает на индикатор.
- BackColor (7f7f00) - позволяет задавать цвет фона.
- TextColor (000000) - позволяет задать цвет текста.
- TextChangeColor (000000) - позволяет устанавливать цвет разряда с измененным значением.
- TextSize (40) - позволяет устанавливать размер текста.
- Enable (true) - позволяет включать и выключать индикатор.
- InactiveColor (c0c0c0) - управляет цветом в неактивном состоянии.
- precision (0.01) - позволяет устанавливать точность отображения.
- digits (3) - позволяет устанавливать число цифр до запятой.
- BoldFont (true) - устанавливает и отключает жирный шрифт.

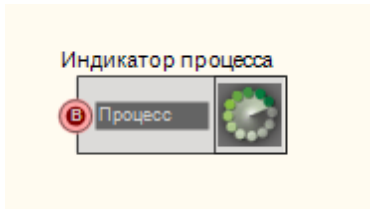

**Пример**

*Пример* создания измерителя переменного сигнала: (находится в Примере компонента [Вольтметр переменного тока](#)<sup>[284]</sup>)

## 12.4. Индикатор процесса

**Индикатор процесса** - компонент. Предназначен для графического отображения бинарного состояния протекания процесса.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Процесс - вывод, к которому подключается канал с бинарными значениями-состояниями.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

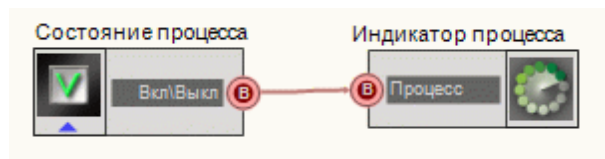
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- process (0) - значение состояния процесса.



## Пример

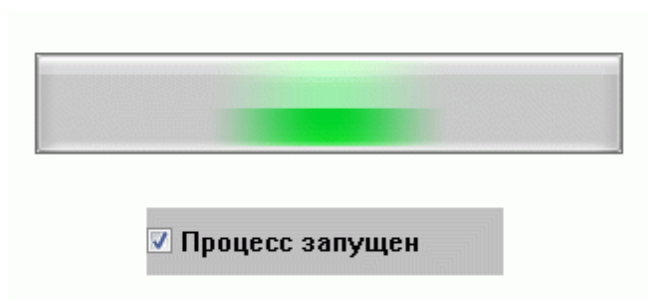
### Результат работы проекта:



При нажатом состоянии компонента [CheckBox](#)<sup>[363]</sup> на экране монитора (в режиме оператора) Индикатор процесса становится видимым и появляется полоса протекания процесса.

### Результат работы проекта





Данный пример непосредственно в ZETView

## 12.5. Индикатор уровня

**Индикатор уровня** - компонент. Предназначен для графического отображения интегрального уровня сигнала. Работает по принципу [Интегратора](#)<sup>[141]</sup>, то есть указывает уровень, равный сумме всех предыдущих значений и текущего.

Принимает значение до 1.

### Внешний вид компонента:

Режим проектировщика	Режим оператора

### Параметры:

#### ➤ Входные:

- Уровень - вывод, к которому подключается канал, в котором требуется измерить уровень сигнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

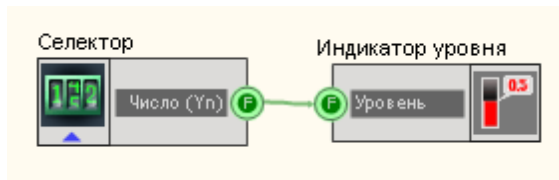
Частные свойства (в скобках значение, установленное по умолчанию):

- Enable (true) - устанавливается активность/неактивность.
- InactivColor (c0c0c0) - устанавливается цвет в неактивном состоянии.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Индикатор уровня](#)<sup>[351]</sup> показывает, уровень сигнала, поступающего с [Селектора](#)<sup>[215]</sup>.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 12.6. Колба

**Колба** - компонент. Предназначен для графического отображения данных в виде индикатора колбы.

### Внешний вид компонента:

Режим проектировщика	Режим оператора

### Параметры:

➤ Входные:

- Включен\Выключен - входной контакт, принимающий число, впоследствии отображаемое на индикаторе.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

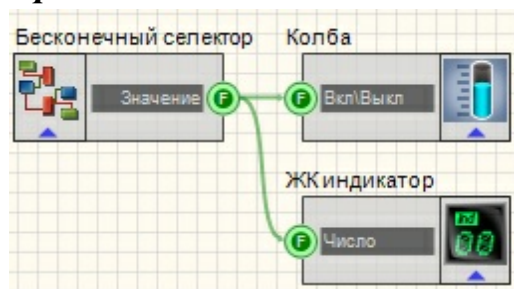
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- TextSize (8) - размер текста.
- textstyle (FontStyleItalic) - стиль текста
- TextColor (000000) - цвет текста.
- CurrentValue (0) - установка текущего значения.
- DigitFrom (0) - начальное значение.
- DigitTo (10) - конечное значение.
- accuracy (0.1) - точность измерений.
- LinearsUnderline (true) - подчёркивание линейки
- LinearVisible (true) - видимость линейки
- linearorientation (left)- сторона отображения линейки:
  1. left - левая сторона.
  2. right - правая сторона.
- BackColorUp (000000) - верхний цвет фона.
- BackColorDown (808080) - нижний цвет фона.
- LiquidUpColor (00f4ff) - верхний цвет градиента курсора.
- LiquidDownColor (00f7ff)- нижний цвет градиента курсора.
- Enabled (true) - включение/отключение компонента.
- AnimationEnabled (true) - включение/отключение анимации.



### Пример

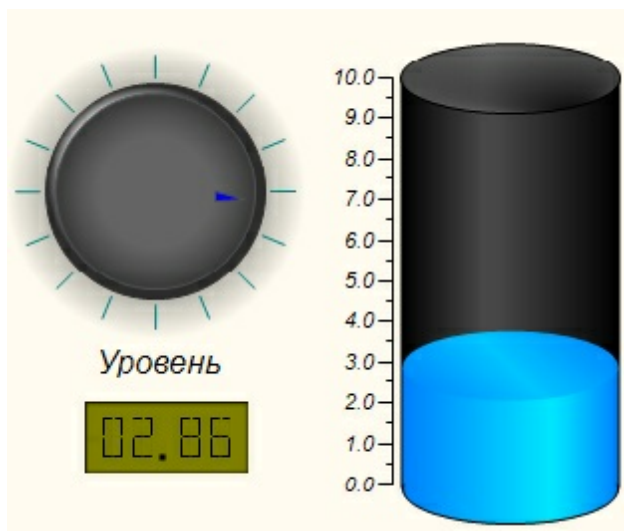
#### Проект в SCADA ZETView



В этой схеме компонент [Бесконечный селектор](#)<sup>[218]</sup> служит для подключения индикатора [Колба](#)<sup>[352]</sup> и позволяет нам

увеличивать или уменьшать уровень в колбе. [Селектор<sup>\[215\]</sup>](#) позволяет нам определить точное значение уровня колбы.

### Результат работы проекта


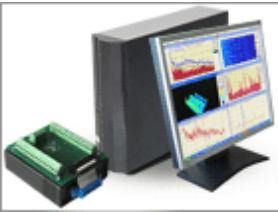


Данный пример непосредственно в ZETView

## 12.7.Рисунок

**Рисунок** - компонент. Предназначен для организации выбора конкретного изображения из набора в зависимости от значения, поступающего на вход.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Номер - контакт, на который передается номер изображения, которое нужно вывести на дисплей в режиме оператора.

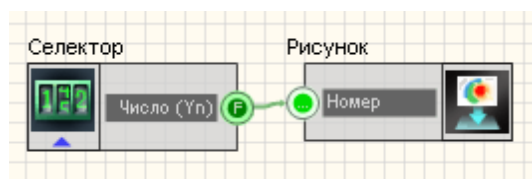
**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- CurrentPicture (0) - установка текущего номера изображения.
- Pic1\_FileName - выбор изображения, которое будет отображаться на дисплее в режиме оператора при подаче на вход компонента числа 1.
- ...
- Pic10\_FileName - выбор изображения, которое будет отображаться на дисплее в режиме оператора при подаче на вход компонента числа 10.

**Пример****Проект в SCADA ZETView**

Выбор рисунков для каждого из значений производится путем редактирования частных свойств компонента:

Частные свойства	
CurrentPicture	2
Pic1_FileName	PIC1.gif
Pic2_FileName	PIC2.gif
Pic3_FileName	PIC3.gif
Pic4_FileName	
Pic5_FileName	
Pic6_FileName	
Pic7_FileName	
Pic8_FileName	
Pic9_FileName	
Pic10_FileName	

В этой схеме компоненты Рисунок служат для вывода изображений на дисплей, которые в зависимости от введенного числа. Компонент [Селектор](#)<sup>[213]</sup> служит для выбора конкретного рисунка.

## Результат работы проекта



Данный пример непосредственно в ZETView

### 12.8.Световой индикатор

**Световой индикатор** - компонент. Предназначен для графического отображения двоичных значений состояния входа в виде индикатора, меняющего цвет.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора

#### Параметры:

##### ➤ Входные:

- Включен\Выключен - вывод, к которому подключается канал с бинарными значениями.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- onColor (00ff00) - цвет индикатора во включенном состоянии.
- offColor (ff0000) - цвет индикатора во выключенном состоянии.
- Enable (true) - позволяет управлять общим состоянием индикатора (включен/

выключен).

- DisableColor (c0c0c0) - позволяет установить цвет при неактивном состоянии.
- currentStatus (true) - позволяет установить текущее состояние индикатора.
- status (true) - то же самое, что и currentStatus.
- figure (Круг) - позволяет устанавливать геометрическую форму индикатора в режиме оператора:
  1. Круг.
  2. Квадрат.



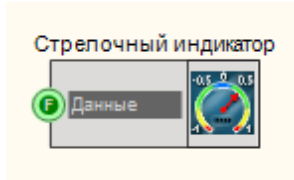

### Пример

*Пример* создания индикатора состояния входа: (находится в Примере компонента [Цифровой порт](#) (609))

## 12.9.Стрелочный индикатор

**Стрелочный индикатор** - компонент. Предназначен для графического отображения данных в виде индикатора стрелочного типа.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

➤ Входящие:

- Данные - вывод, к которому подключается канал с численными значениями, которые необходимо отобразить.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Max (10) - максимальное значение, которое сможет отобразить индикатор.
- DivCount (10) - общее число делений на шкале.
- Value (0) - текущее значение.
- TextSize (9) - размер текста цифр на индикаторе.
- textStyle (FontStyleBoldItalic) - стиль текста цифр на индикаторе.
- BackColor (0000ff) - цвет фона и ободов индикатора.
- ArrowColor (ff0000) - цвет стрелки.
- ArrowWidth (4) - толщина стрелки.
- haveArrow (false) - включение/выключение формы указателя в виде стрелки.
- digits (0.1) - установка точности.
- Enable (true) - активация/деактивация работы индикатора.
- Amplifier (1) - установка множителя шкалы.
- Unit (мВ) - единица измерения.

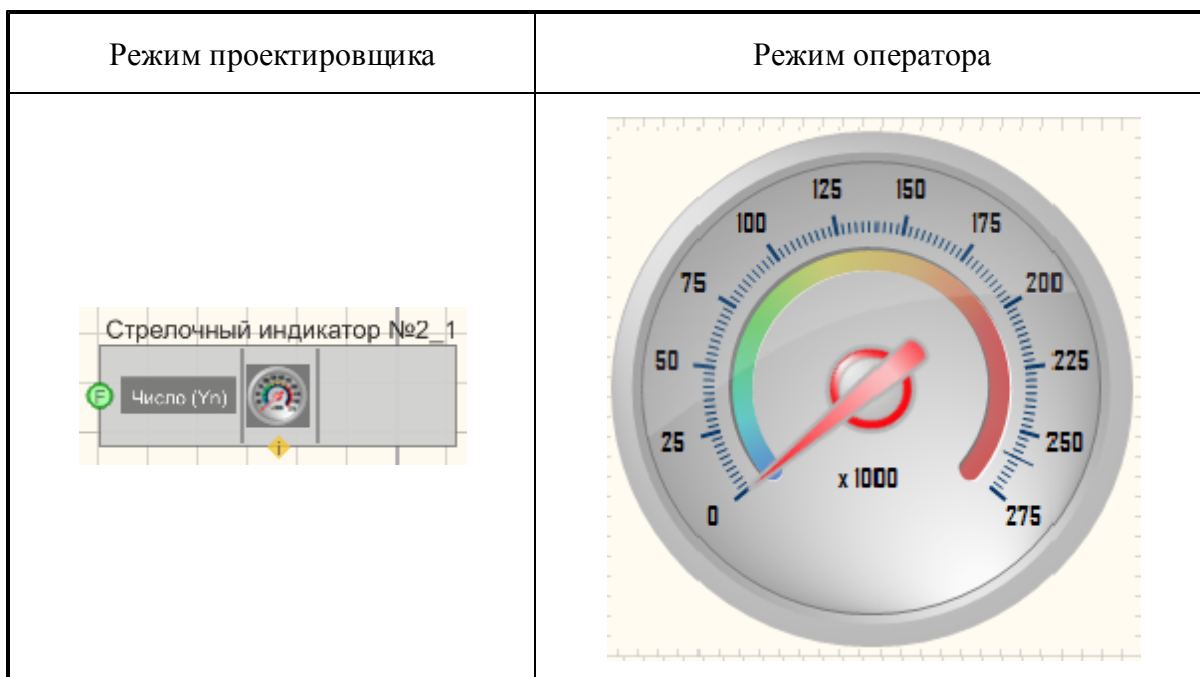
---

## 12.10. Стрелочный индикатор №2

**Стрелочный индикатор №2** - компонент. Предназначен для графического отображения данных в виде индикатора стрелочного типа.

**Внешний вид компонента:**





### Параметры:

#### ➤ Входные:

- Число (Yn) - вывод, к которому подключается канал с численными значениями, которые необходимо отобразить.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

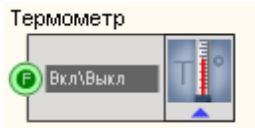
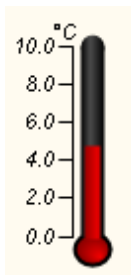
- CurrentValue (0) - текущее значение;
- style (объемный) - стиль (объемный/плоский);
- DrawBorder (false) - отображение рамки (Нет/Да);
- BorderColor (003366) - цвет рамки;
- GaudColor (b9b9b9) - цвет внешней части прибора;
- DialBackColor (a5a5a5) - цвет циферблата;
- DigitsColor (000000) - цвет чисел;
- BigLinesColor (003366) - цвет больших рисок;
- LittleLinesColor (003366) - цвет маленьких рисок;
- NeedleColor (fa000a) - цвет стрелки;
- UnderlineColor (fafa0a) - цвет подчеркивающей линии;
- UnderlineVisible (false) - отображение подчеркивающей линии (Нет/Да);
- DigitsVisible (true) - отображение чисел (Нет/Да);
- BigLinesLength (10) - длина больших рисок;

- LittleLinesLength (5) - длина маленьких рисок;
- BigLinesWidth (2) - ширина больших рисок;
- LittleLinesWidth (1) - ширина маленьких рисок;
- UnderlineWidth (1) - ширина подчеркивающей линии;
- NeedleLength (74)- длина стрелки;
- needlewidth (6) - параметр, показывающий, во сколько раз ширина половины стрелки меньше ширины круглого основания стрелки;
- NeedleArrowWidth (1) - ширина кончика стрелки;
- AngleFrom (-40) - угол, от которого начинается шкала;
- AngleTo (260) - угол до которого размечается шкала;
- ValueFrom (0) - значение в начальной точке;
- ValueTo (260) - значение в конечной точке;
- Multiplexor (1000) - число, отображаемое под стрелкой
- fontstyle (FontStyleBold) - стиль шрифта;
- FontSize (10) - размер шрифта;
- WidthZoneLine (10) - ширина линии, которая отображает зоны;
- RedZonePercent (30) - процентное значение участка красной зоны;
- YellowZonePercent (25) - процентное значение участка желтой зоны;
- GreenZonePercent (45) - процентное значение участка зеленой зоны;
- GradientZoneColorTone (a5a5a5) - тон линии, которая отображает зоны при установленном стиле GDT\_APART\_GRADIENT
- digitspos (числа над рисками) - расположение чисел относительно рисок (числа над рисками/числа под рисками);
- llvs (10) - частота расположения маленьких рисок;
- zonedialtype (отдельно градиентная) - тип линии, которая отображает зоны:
  1. отсутствует;
  2. отдельно градиентная;
  3. отдельно четные границы;
  4. отдельно только красная;
  5. на рисках;
  6. на рисках только красная;
- needlestyle (Стиль\_2) - стиль стрелки (от Стиль\_1 до Стиль 7).

## 12.11.Термометр

**Термометр** - компонент. Предназначен для графического отображения данных в виде индикатора термометра.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Включен\Выключен - входной контакт, принимающий численное значение, которое будет отображено на термометре.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

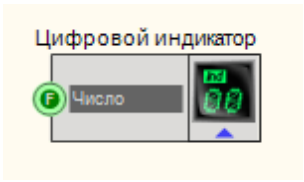
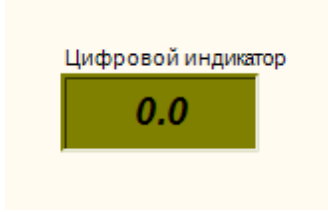
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию)

- TextSize (8) - размер текста.
- textstyle (FontStyleItalic) - стиль текста
- TextColor (000000) - цвет текста.
- CurrentValue (0) - установка текущего значения.
- DigitFrom (0) - начальное значение.
- DigitTo (10) - конечное значение.
- accuracy (0.1) - точность измерений.
- LinearUnderline (true) - подчёркивание линейки
- LinearVisible (true) - видимость линейки
- linearorientation (left) - сторона отображения линейки:
  1. left - левая сторона.
  2. right - правая сторона.
- BackColorUp (000000) - верхний цвет фона.
- BackColorDown (808080) - нижний цвет фона.
- LiquidUpColor (f40000) - верхний цвет градиента курсора.
- LiquidDownColor (7f0000) - нижний цвет градиента курсора.
- Enabled (true) - включение/отключение компонента.
- measuring (град) - позволяет установить единицы измерения.
- AnimationEnabled (true) - включение/отключение анимации.

## 12.12.Цифровой индикатор

**Цифровой индикатор** - компонент. Предназначен для графического отображения данных в виде цифрового индикатора.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Число - вывод, к которому подключается канал с численными значениями, которые необходимо отобразить.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IndValue (0) - отображаемое значение.
- BackColor (7f7f00) - фоновый цвет.
- TextColor (000000) - цвет текста.
- TextSize (14) - размер текста цифр.
- textstyle (FontStyleItalic) - стиль текста цифр.
- digits (0.1)- установка точности.
- Enable (true) - активация/деактивация работы индикатора.
- InactiveColor (c0c0c0) - цвет в неактивном режиме.
- valuetype (Десятичный) - формат отображения числа:
  1. Восьмиричный.
  2. Десятичный
  3. Шестнадцатиричный



### *Пример*

*Пример* создания индикатора состояния входа: (находится в Примере

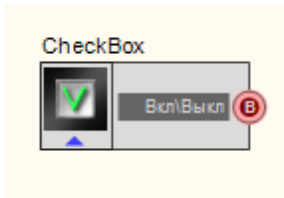

компонента [Вольтметр постоянного тока](#)<sup>[288]</sup>)

## Глава 13. Кнопки

### 13.1. CheckBox

**CheckBox** - компонент. Используется для предоставления пользователю выбора между дискретными состоянием Истина (true) и Ложно (false).

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	 <p>1- Флаг снят                      2- Флаг установлен</p>

#### Параметры:

##### ➤ Выходные:

- Вкл\Выкл - вывод, на котором выставляются два логических состояния, которые зависят от того, выставлен флаг или нет.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

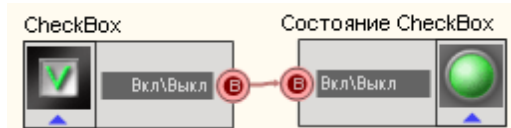
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Check (false) - состояние компонента.
- Name (Check) - текст управляющего компонента.
- Color (c0c0c0) - цвет фона.
- Transparent (false) - включение/выключение прозрачности.



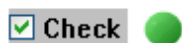
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Световой индикатор](#)<sup>[356]</sup> служит для графического представления состояния **CheckBox**.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 13.2. Кнопка с картинкой

**Кнопка с картинкой** - компонент. Предназначена для установки в проект кнопки, которая, по сути является кнопкой с фиксацией, но вместо надписи имеет картинку.

### Внешний вид компонента:

Режим проектировщика	Режим оператора

### Параметры:

#### ➤ Выходные:

- Импульс - вывод, на котором при нажатии кнопки возникнет кратковременно меняющийся логический уровень.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- `imagesource` (загрузить из списка) - источник загрузки изображения (загрузить из списка/загрузить из файла).
- `image` (Очистить) - выбор внешнего вида кнопки на панели оператора:

1. Калькулятор.
  2. Часы.
  3. Дискета.
  4. Документ.
  5. Параметры.
  6. Принтер.
  7. Конфигурация.
  8. Вибростенд 1.
  9. Вибростенд 2.
  10. Информация.
  11. Таймер.
  12. Обновление.
  13. Выбор папки.
  14. Сохранить.
  15. Открыть.
  16. Вперед.
  17. Назад.
  18. Вверх.
  19. Старт.
  20. Стоп.
  21. Очистить.
- FileName - установка названия файла загружаемого изображения и местоположения на диске. Позволяет загружать свои рисунки.
  - ToolTip - текст всплывающей подсказки.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Световой индикатор](#)<sup>[356]</sup> служит для графического представления состояния последней нажатой [Кнопки\\_с картинкой](#)<sup>[364]</sup>.

### Результат работы проекта



Данный пример непосредственно в ZETView

### 13.3.Кнопка с фиксацией

**Кнопка с фиксацией** - компонент. Кнопка фиксирует свое состояние. Передает значение "1" при переходе в состояние "Включено", "0" - при переходе в состояние "Выключено".

#### Внешний вид компонента:

Режим проектировщика	Режим оператора

#### Параметры:

##### ➤ Выходные:

- Вкл\Выкл - вывод, на котором выставляются два логических состояния, которые зависят от того, нажата кнопка или нет.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- status (false) - текущее состояние компонента. Если параметр True, то при запуске проекта кнопка будет уже нажатой.
- TextSize (12) - размер текста.
- textstyle (FontStyleRegular)- стиль текста.
- OnBackColor (00ff00) - цвет во включенном состоянии.



- OffBackColor (ff0000)- цвет в выключенном состоянии.
- Enable (true) - активация/деактивация компонента.
- OnCaption (Вкл.)- название во включенном состоянии.
- OffCaption (Выкл.)- название в выключенном состоянии.
- AnimationEnable (false)- включение/отключение анимации.



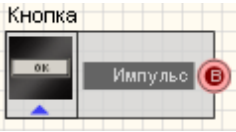

### Пример

Пример создания индикатора состояния входа: (находится в Примере компонента [Цифровой порт](#)<sup>[609]</sup>)

## 13.4.Кнопка без фиксации

**Кнопка без фиксации** - компонент. Кнопка, которая после нажатия возвращается в исходное состояние. При нажатии кнопка передает импульс "1", затем возвращается в "0".

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Импульс - вывод, на котором кратковременно изменяется логический уровень при нажатии (низкий->высокий->низкий).

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

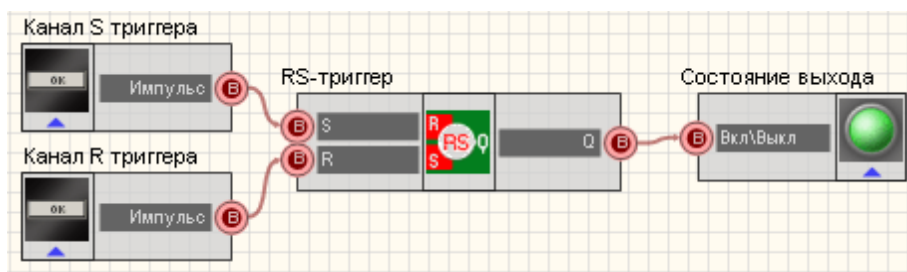
- TextSize (12)- размер текста.
- textStyle (FontStyleRegular) - стиль текста.
- Enable (true) - активация и деактивация кнопки.
- OnCaption (кнопка) - название кнопки во включенном состоянии.

- BitColor (46d6e1) - цвет кнопки в запущенном состоянии проекта, при наведении мышью на нее.
- AnimationEnable (true) - включение/отключение анимации.



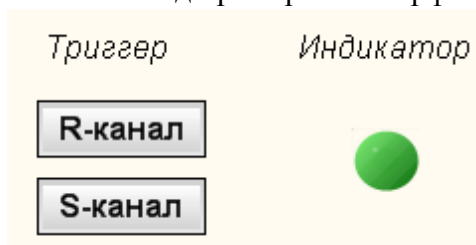
## Пример

### Проект в SCADA ZETView



В этой схеме компоненты [Кнопка без фиксации](#)<sup>[367]</sup> служат для изменения состояний на входах [RS-триггера](#)<sup>[386]</sup>. Компонент [Световой индикатор](#)<sup>[356]</sup> служит для отображения выхода триггера. При последовательном нажатии [Кнопки без фиксации](#)<sup>[367]</sup> меняется состояние [Светового индикатора](#)<sup>[356]</sup>.

Внешний вид примера в интерфейсе оператора.

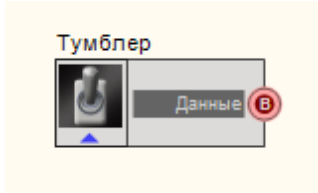
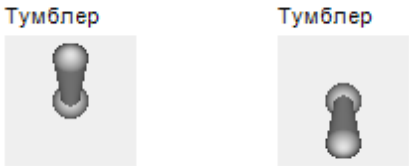


Данный пример непосредственно в ZETView

## 13.5.Тумблер

**Тумблер** - компонент. Предназначен для установки в проект переключателя типа тумблер на два положения, которым соответствуют высокий и низкий логические уровни.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Данные - вывод, на котором выставляются два различных логических состояния, которые зависят от того, в каком положении находится ручка переключения тумблера.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Color (505050) - цвет ручки переключателя тумблера.
- Status (false) - положение тумблера (включен (true)/выключен (false))



## Пример

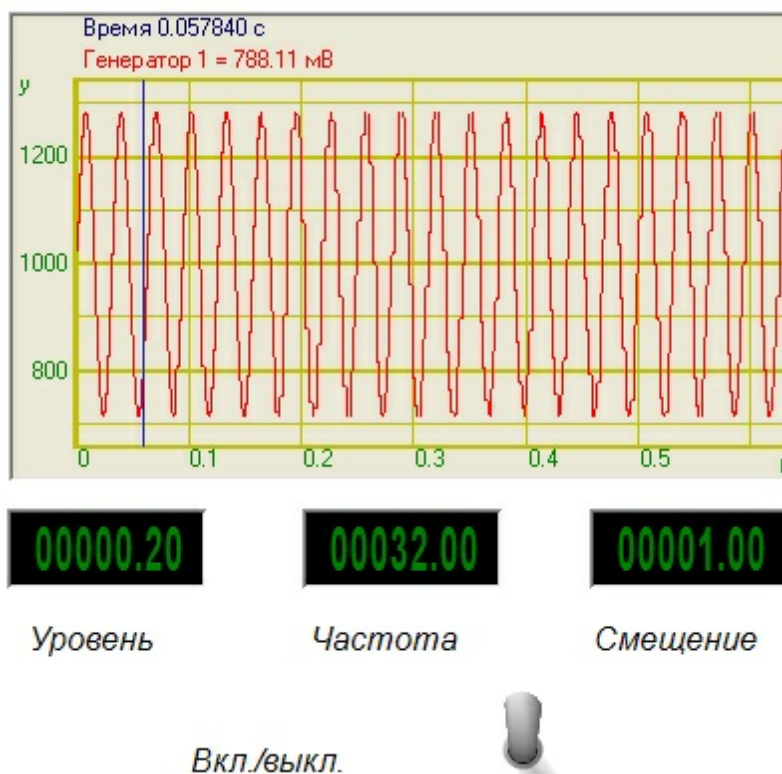
### Проект в SCADA ZETView



В этой схеме компонент [Синусоидальный сигнал](#)<sup>[247]</sup> создает

синусоидальный сигнал, с параметрами, которые задаются при помощи [Селекторов](#)<sup>[215]</sup>. [Тумблер](#)<sup>[368]</sup> позволяет включать и выключать [Многоканальный осциллограф](#)<sup>[494]</sup> с [Синусоидальный сигнал Генератора сигналов](#)<sup>[247]</sup>. Для просмотра сгенерированного сигнала служит компонент [Многоканальный осциллограф](#)<sup>[494]</sup>. Результат работы составленной программы можно увидеть в интерфейсе оператора.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## Глава 14. Конвертация

### 14.1. Конвертер (логика - строка)

**Конвертер (логика - строка)** - компонент. Предназначен для перевода данных из булевого типа в текстовые сообщения.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной канал, на который поступают данные булевого типа.
- Вкл./Выкл. - вход для кнопки включения.

#### ➤ Выходные

- Выход - выходной канал, который в зависимости от полученного на входе логического уровня, на выходе формирует текстовое сообщение.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- TrueString (Логическая 1) - сообщение, которое будет выведено в случае поступления на вход логической единицы.
- FalseString (Логический 0) - сообщение, которое будет выведено в случае поступления на вход логического нуля.
- RepeatData (true) - включение выключение на повторное сообщение.
- Enable (true) - установка активного/неактивного состояния компонента.
- InactiveText (Компонент в неактивном состоянии) - сообщение, которое будет выведено при переходе в неактивное состояние.
- ActiveText (Компонент в активном состоянии) - сообщение, которое будет выведено при переходе в активное состояние.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Конвертер \(логика - строка\)](#)<sup>370</sup> в

зависимости от поступающего на вход логического уровня на выходе передает заданное текстовое сообщение. [Кнопка без фиксации](#)<sup>[367]</sup>, подключенная к [Менеджеру свойств](#)<sup>[534]</sup> служит для активации/деактивации [Конвертера \(логика - строка\)](#)<sup>[370]</sup>. [Метка](#)<sup>[554]</sup> необходима для графического отображения сообщений. [Менеджер свойств](#)<sup>[534]</sup> предназначен для открытия доступа к частному свойству Enable (компонента [Конвертер \(логика - строка\)](#)<sup>[370]</sup>). [Кнопка без фиксации](#)<sup>[367]</sup>, подключенная к [Конвертера \(логика - строка\)](#)<sup>[370]</sup>, нужна для изменения логических уровней на входе.

### Результат работы проекта

Логическая 1

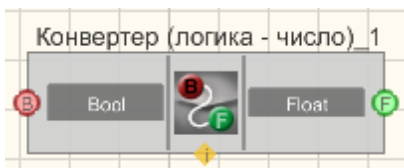
Высокий  
уровень

Данный пример непосредственно в ZETView

## 14.2. Конвертер (логика - число)

**Конвертер (логика - число)** - компонент. Предназначен для перевода данных из булевого типа в числовой.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Bool - входной канал, на который поступают данные булевого типа.

#### ➤ Выходные

- Float - выходной канал, который выдает поставленные в соответствие с булевыми значениями действительные числа.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- TrueConversion (1) - установка действительного числа, соответствующего истине (единица в булевой алгебре).
- FalseConversion (0) - установка действительного числа, соответствующего ложному значению (ноль в булевой алгебре).
- RepeatData (true) - установка реакции на повторное сообщение (true/false)



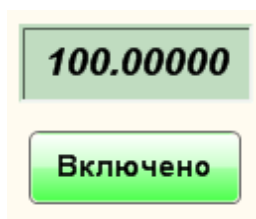
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Конвертер \(логика - число\)](#)<sup>[372]</sup> переводит состояния [Кнопки с фиксацией](#)<sup>[366]</sup> (передает двоичные состояния) в действительные значения, заданные пользователем в настройках, в данном примере это 100 при высоком логическом уровне на входе и -250 при низком. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления численных значений.

### Результат работы проекта



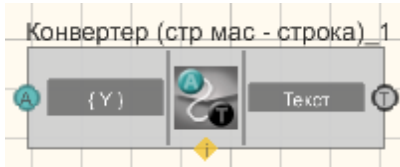
Данный пример непосредственно в ZETView

## 14.3.Конвертер (стр.масс. - строка)

**Конвертер (стр.массив - строка)** - компонент. Предназначен для перевода текстовых массивов в строку.

Когда на вход приходят неконвертируемые символы, компонент передает на выход конвертируемую строку.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- {Y} - Входной строковый массив.

## ➤ Выходные:

- Текст - Выходная конвертируемая строка.

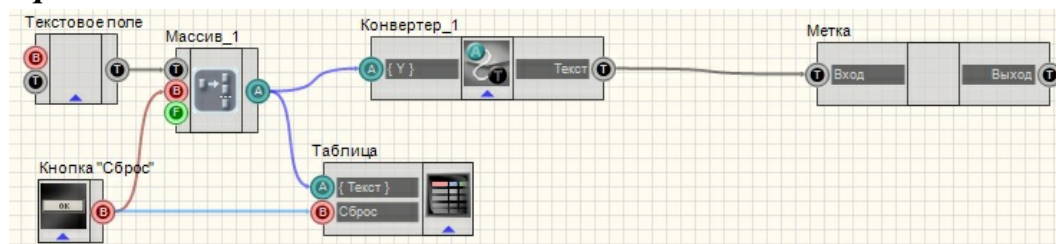
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства:

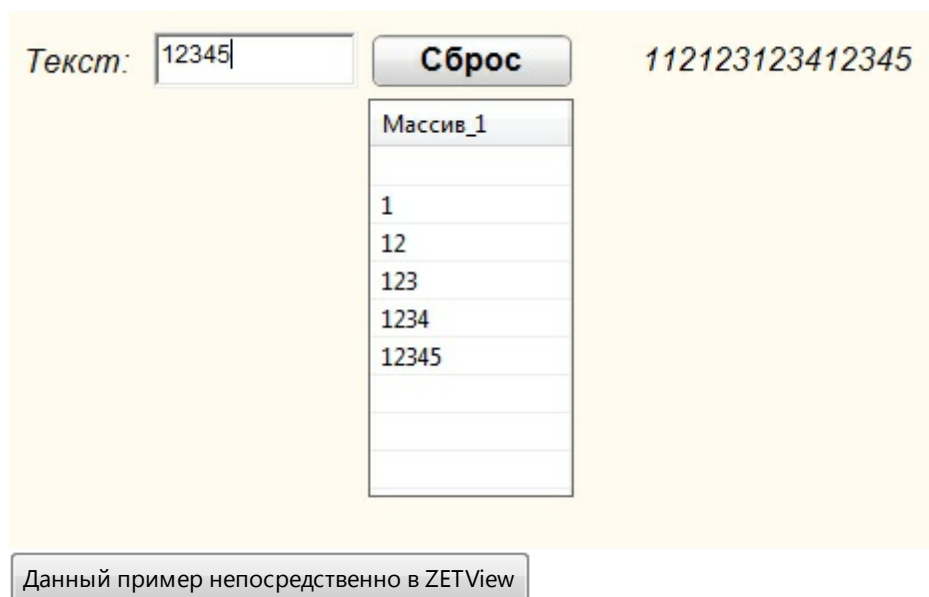
Не имеет.

**Пример****Проект в SCADA ZETView**

В представленной схеме показана работа компонента "[Конвертер \(стр.мас.-строка\)](#)".<sup>[373]</sup> Строковый массив на входе компонента формируется следующим путем: в [текстовое поле](#)<sup>[561]</sup> пользователь вводит числа, которые поступают на компонент [Формировщик массивов \(текст-массив\)](#)<sup>[422]</sup>, далее массив выводится в [Таблицу данных](#)<sup>[496]</sup> и попадает на [конвертер \(стр.мас.-строка\)](#)<sup>[373]</sup>. Полученные данные выводятся на метку. [Кнопка без фиксации](#)<sup>[367]</sup> используется в данном проекте, чтобы сбросить массив и значения в таблице.



## Результат работы проекта



### 14.4.Конвертер (стр.мас. - числ.мас.)

**Конвертер (стр.массив-числ.массив)** - компонент. Предназначен для перевода текстовых массивов в числовые.

Когда на вход приходят неконвертируемые символы, компонент передает их на выход как число 0.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- {Y} - Входной строковый массив.

##### ➤ Выходные:

- {Y} - Выходной числовой массив.

#### Настраиваемые свойства:

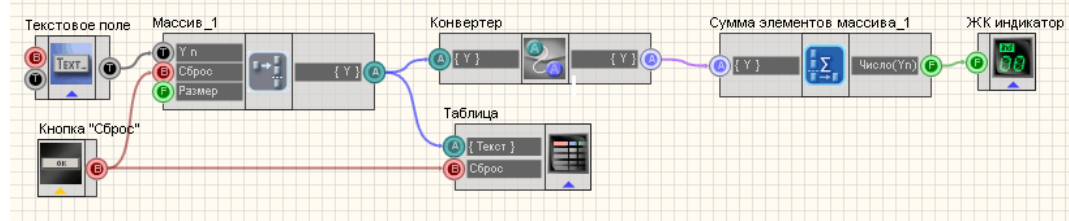
##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>  
➤ Частные свойства:  
Не имеет.



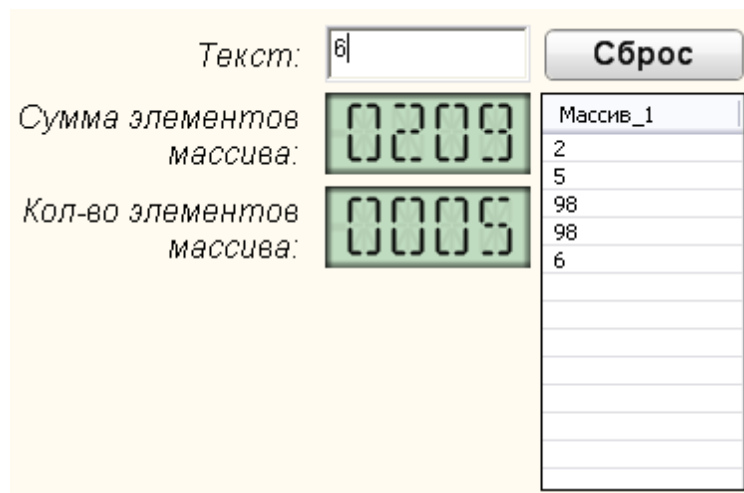
## Пример

### Проект в SCADA ZETView



В представленной схеме показана работа компонента "[Конвертер \(стр.мас.-числ.мас.\)](#)".<sup>[373]</sup> Строковый массив на входе компонента формируется следующим путем: в [текстовое поле](#)<sup>[561]</sup> пользователь вводит числа, которые поступают на компонент [Формировщик массивов \(текст-массив\)](#)<sup>[422]</sup>, далее массив выводится в [Таблицу данных](#)<sup>[496]</sup> и попадает на [конвертер \(стр.мас.-числ.мас.\)](#)<sup>[373]</sup>, где из строкового массива преобразуется в числовой. Далее все элементы полученного числового массива складываются с помощью компонента [Сумма элементов массива](#)<sup>[448]</sup> и результат выводится на [ЖК индикатор](#)<sup>[348]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> используется в данном проекте, чтобы сбросить массив и значения в таблице.

### Результат работы проекта



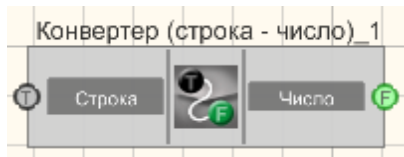
Данный пример непосредственно в ZETView

## 14.5.Конвертер (строка - число)

**Конвертер (строка - число)** - компонент. Предназначен для перевода чисел, представленных в текстовом типе, в числовой.

Когда на вход приходят неконвертируемые символы, компонент передает их на выход как число 0.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Строка - входной канал, на который поступают текстовые данные.

#### ➤ Выходные

- Число - выходной канал, который передает преобразованный в численный тип текстовые значения.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

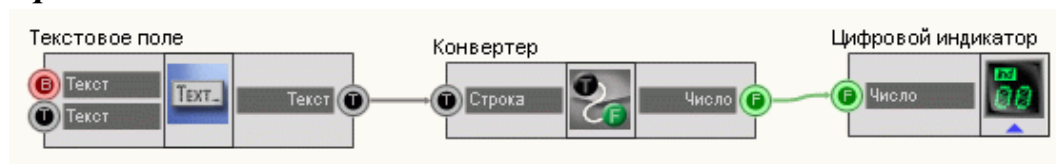
#### Частные свойства:

Не имеет.



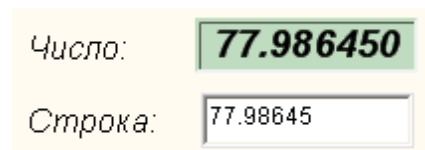
## Пример

### Проект в SCADA ZETView



В этой схеме компонент Конвертер (строка - число) переводит значения, представленные в текстовом виде, в числовой тип. [Цифровой индикатор](#)<sup>[362]</sup> служит для графического представления информации. [Текстовое поле](#)<sup>[561]</sup> предназначено для ввода чисел в текстовой форме.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 14.6. Конвертер (числ.массив.-стр.массив.)

**Конвертер (числ.массив-стр.массив)** - компонент. Предназначен для перевода числовых массивов в текстовые.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входящие:

- {Y} - Входной числовой массив.

#### ➤ Выходные:

- {Y} - Выходной строковый массив.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

[Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

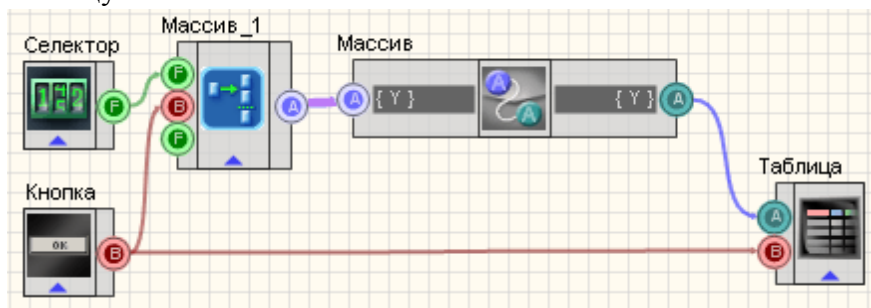
- precision (0) - позволяет устанавливать точность отображения.
- type (Текстовый) - устанавливается тип конвертации (Текстовый/Символьный)



## Пример

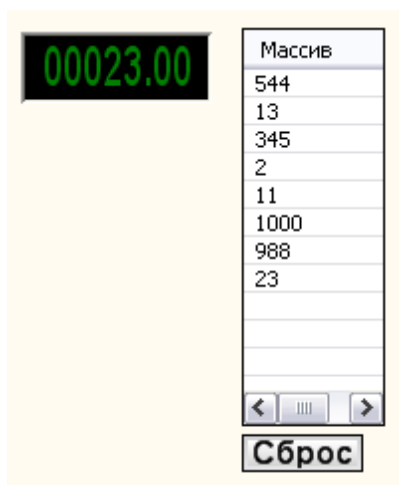
### Проект в SCADA ZETView

Чаще всего данный компонент применяется для вывода результатов в таблицу:



В данной схеме показана работа компонента ["Конвертер\(числ.мас.-стр.мас.\)"](#)<sup>[378]</sup>. Числа с селектора попадают в [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup>, после чего на компоненте ["Конвертер\(числ.мас.-стр.мас.\)"](#)<sup>[378]</sup> происходит преобразование числового массива в текстовый, что позволяет вывести результат в [таблицу данных](#)<sup>[496]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> используется в данном проекте чтобы сбросить массив и значения в таблице.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 14.7. Конвертер (число - строка)

**Конвертер (число - строка)** - компонент. Предназначен для перевода данных из численного типа в строковый.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Число - входной канал, на который поступают данные численного типа.

## ➤ Выходные

- Строка - выходной канал, который передает преобразованный в строковый тип численные значения.

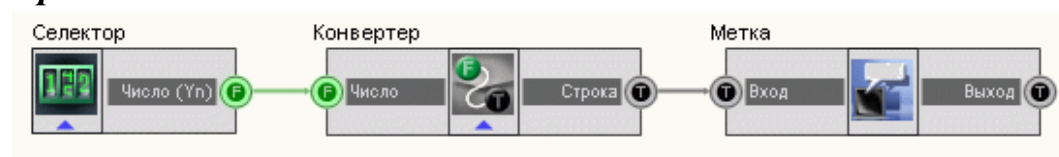
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- `тире (Десятичный)` - установка типа конвертации (происходит преобразование из одной системы счисления в другую перед преобразованием в строковый тип):
  1. Восьмеричный.
  2. Десятичный.
  3. Шестнадцатеричный.
- `RepeatData (true)` - установка реакции на повторяющееся сообщение. Если значение True, компонент будет конвертировать все сообщения приходящие на вход, если False, то при получении повторяющегося сообщения компонент проигнорирует его, действует только для сообщений идущих подряд:  
 10,12,10,13,14 будут конвертированы как 10,12,10,13,14  
 10,12,10,10,14 будут конвертированы как 10,12,10,14

**Пример****Проект в SCADA ZETView**

В этой схеме компонент "[Конвертер \(число - строка\)](#)"<sup>[379]</sup> переводит численные значения, полученные с [Селектора](#)<sup>[213]</sup>, в строковый тип. [Метка](#)<sup>[554]</sup> служит для графического отображения полученной текстовой информации.

### Результат работы проекта

440.230011

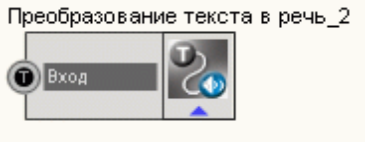
00440.23

Данный пример непосредственно в ZETView

## 14.8.Преобразование текста в речь

**Преобразование текста в речь** - компонент. Предназначен для озвучивания текстовой информации.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной канал, на который поступают текст, который в дальнейшем будет преобразован в речь.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства:

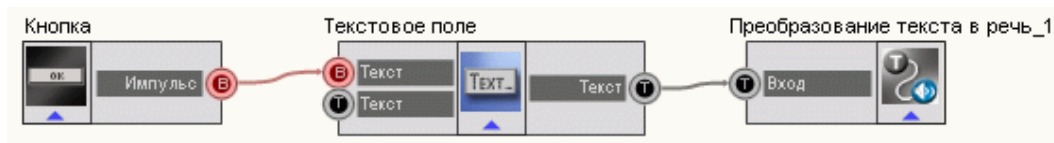
- SpeechText - вводится текст, который будет озвучен.
- SpeechEngine - указывается голосовой модуль, который будет использован при озвучке (чтобы подключить голосовой модуль, необходимо выбрать компонент, далее вызвать контекстное меню, в котором выбрать пункт Редактировать компонент.

Появится список голосовых модулей, установленных в системе, необходимо выбрать один из них).



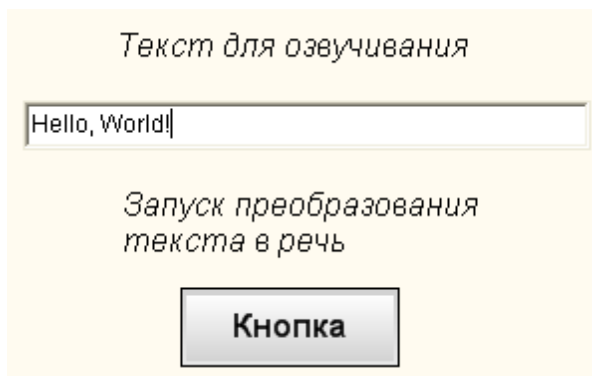
## Пример

### Проект в SCADA ZETView



В этой схеме компонент ["Преобразование текста в речь"](#)<sup>[381]</sup> служит для преобразования текстовых сообщений в звуковые. [Текстовое поле](#)<sup>[561]</sup> требуется для ввода текста в компонент.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 14.9. Преобразование цвета

**Преобразование цвета** - компонент. Предназначен для перевода полученных числовых значений параметров RGB в RGB- код.

RGB (аббревиатура английских слов Red, Green, Blue — красный, зелёный, синий) — аддитивная цветовая модель, как правило, описывающая способ синтеза цвета для цветопроизведения, каждый цвет представлен в виде комбинации красного (R - red), зеленого (G - green) и синего (B - blue) цветов в определённых пропорциях.



**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Red - Интенсивность красной составляющей цвета (от 0 до 255).
- Green - Интенсивность зеленой составляющей цвета (от 0 до 255).
- Blue - Интенсивность синей составляющей цвета (от 0 до 255).

## ➤ Выходные:

- Цвет - Выходное значение цвета цветовой модели RGB. Результат представлен в виде кода из 6 цифр в десятичной системе счисления.

**Настраиваемые свойства:**

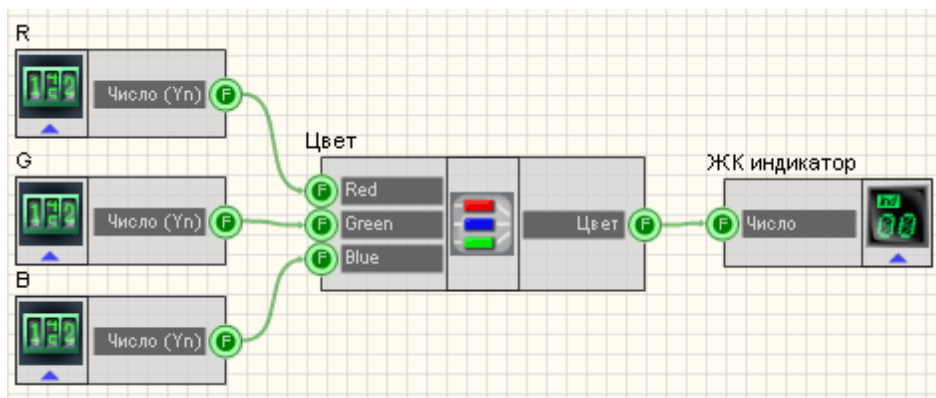
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

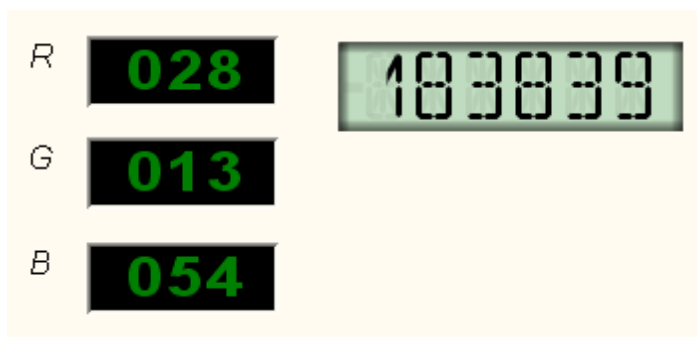
- Red (0) - Значение параметра красного цвета на входе компонента по умолчанию.
- Green (0) - Значение параметра зеленого цвета на входе компонента по умолчанию.
- Blue (0) - Значение параметра синего цвета на входе компонента по умолчанию.

***Пример******Проект в SCADA ZETView***



Три [селектора](#)<sup>[213]</sup> используются для того чтобы выставить входные параметры RGB. Результат выводится на [ЖК индикатор](#)<sup>[348]</sup>

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## Глава 15. Логика

### 15.1. D-триггер

**D-триггер** - компонент. Предназначен для использования в качестве тактируемого D-триггера. На вход D поступают данные любого типа, но не передаются следующему компоненту, пока на входе B триггера стоит значение 0 (по умолчанию). В момент, когда D-триггер получает управляющий импульс 1 на вход B, он передает последнее полученное значение следующему компоненту. После этого на входе B снова устанавливается 0.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные

- D - контакт, к которому подключается канал данных.
- Синхро - дискретный вход, при подаче на который синхроимпульса, данные с контакта D будут переданы на выход Q.

#### ➤ Выходные:

- Q - выходной вывод, на который будут поступать данные со входа.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

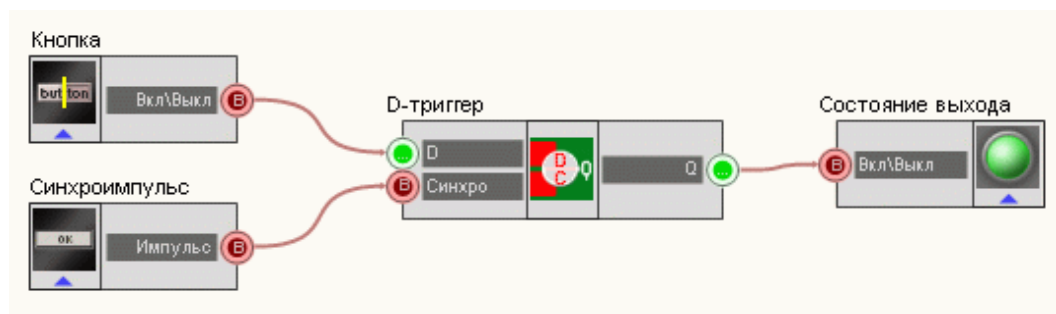
#### ➤ Частные свойства:

Не имеет.



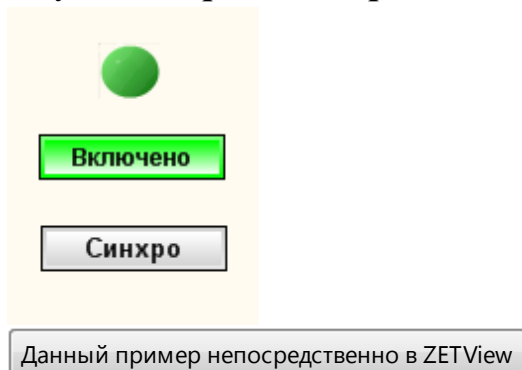
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Кнопка с фиксацией](#)<sup>[366]</sup> служит для установки бинарного состояния, которое будет передано на [Световой индикатор](#)<sup>[356]</sup> при помощи **D-триггера**. [Кнопка без фиксации](#)<sup>[367]</sup> нужна для отправки синхроимпульса на тактируемый вход триггера (чтобы тот передал свое состояние со входа на выход). [Световой индикатор](#)<sup>[356]</sup> нужен для графического отображения дискретных состояний.

### Результат работы проекта



### Математическое описание

Триггер - это электрическая схема, имеющая два устойчивых состояния, которые устанавливаются при подаче соответствующей комбинации сигналов на управляющие входы триггера и сохраняющиеся в течение заданного времени после окончания действия этих сигналов. Триггер - логическое устройство способное хранить 1 бит данных.

В D-триггере при помощи дополнительных логических элементов удалось избежать запрещенной ситуации, когда состояние триггера оказывается неопределенно. Таблица истинности в этом случае имеет вид:

CLK	D	Q	Q	
0	X	Q <sub>-1</sub>	Q <sub>-1</sub>	Режим хранения информации
1	0	0	1	Режим записи информации
1	1	1	0	

## 15.2.RS-триггер

**RS-триггер** - компонент. Предназначен для использования в качестве RS-триггера.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- S - вывод (Set), при подаче на который импульса устанавливает на выходе низкий уровень.
- R - вывод (Reset), при подаче на который импульса устанавливает на выходе высокий уровень.

## ➤ Выходные:

- Q - вывод, на котором в зависимости от входных значений выставляется определенный логический уровень.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства:

Не имеет.

**Пример**

Пример создания индикатора изменения состояния входа на базе триггера: (находится в Примере компонента [Кнопка без фиксации](#)<sup>[367]</sup>)

**Математическое описание**

Триггер - это электрическая схема, имеющая два устойчивых состояния, которые устанавливаются при подаче соответствующей комбинации сигналов на управляющие входы триггера и сохраняющиеся в течение заданного времени после окончания действия этих сигналов. Триггер - логическое устройство способное хранить 1 бит данных.

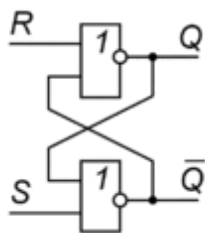


Рис.1

RS-триггер реализуется на основе двух инверторов с перекрестными обратными связями. RS-триггер, показаны на рис.1, состоит из двух элементов ИЛИ-НЕ. Входы триггера обозначаются S (Set - установка) и R (Reset - сброс). Его работа описывается следующей таблицей истинности:

S	R	Q	$\bar{Q}$
0	0	$Q_{-1}$	$\bar{Q}_{-1}$
0	1	0	1
1	0	1	0
1	1	0	0

$Q_{-1}$  – предыдущее состояние

При  $S=0, R=0$  состояние триггера не меняется, этот режим называется режимом хранения информации. При  $S=R=1$  оба выхода триггера должны равняться нулю, что логически невозможно, поэтому, поэтому такое состояние называется неопределенным или запрещенным, и таких ситуаций надо избегать. Режим  $S=1, R=0$ , является режимом записи логической единицы (режимом установки), а режим  $S=0, R=1$  –режимом логического нуля (режимом сброса).

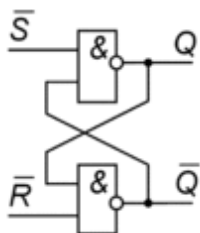


Рис.2

На рис.2 показан RS-триггер с инверсными входами, переключение состояний

которого осуществляется сигналами низкого уровня. Для такого триггера таблица истинности имеет вид:

$\neg S$	$\neg R$	Q	$\neg Q$
0	0	1	1
0	1	1	0
1	0	0	1
1	1	$Q_{-1}$	$\neg Q_{-1}$

$Q_{-1}$  – предыдущее состояние

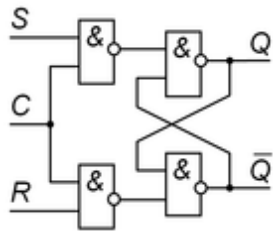


Рис.3

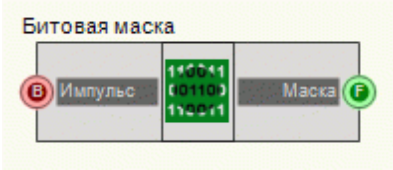
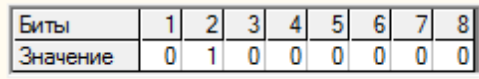
На Рис.3 показана модификация RS-триггера, называемая управляемым или синхронным RS-триггером. Такой триггер меняет свое состояние, только когда на вход синхронизации CLK подан сигнал высокого уровня. При сигнале на этом входе низкого уровня триггер сохраняет свое предыдущее состояние. Для управляемого RS-триггера характерна следующая таблица истинности.

CLK	S	R	Q	$\neg Q$	
1	0	0	$Q_{-1}$	$\neg Q_{-1}$	Как RS-триггер
1	0	1	0	1	
1	1	0	1	0	
1	1	1	?	?	
0	X	X	$Q_{-1}$	$\neg Q_{-1}$	Режим хранения информации

## 15.3. Битовая маска

**Битовая маска** - компонент. Служит для реализации маскирования — выбора и установки отдельных битов или наборов из нескольких битов из двоичной строки или числа (позволяет осуществлять управление устройствами, имеющих 2 состояния).

### Внешний вид компонента:

Режим проектировщика	Режим оператора																		
	 <table border="1"> <thead> <tr> <th>Биты</th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> <th>8</th> </tr> </thead> <tbody> <tr> <td>Значение</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table>	Биты	1	2	3	4	5	6	7	8	Значение	0	1	0	0	0	0	0	0
Биты	1	2	3	4	5	6	7	8											
Значение	0	1	0	0	0	0	0	0											

### Параметры:

#### ➤ Входные:

- Импульс - входной канал, при подаче на который синхроимпульса на выходе формирует выставленную маску.

#### ➤ Выходные

- Маска - выходной канал, по которому происходит отсылка установленной битовой маски.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

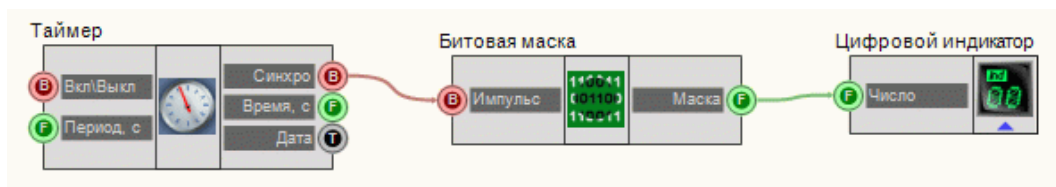
- Mask (1) - текущее десятичное значение маски.
- BitQuantity (1) - установка количества разрядов маски.
- FalseValue (0) - обозначение логического нуля.
- TrueValue (1) - обозначение логической единицы.
- Bitsname (Биты) - установка названия строки с битами.
- ValuesName (Значение) - установка названия строки значения битов.



### *Пример*

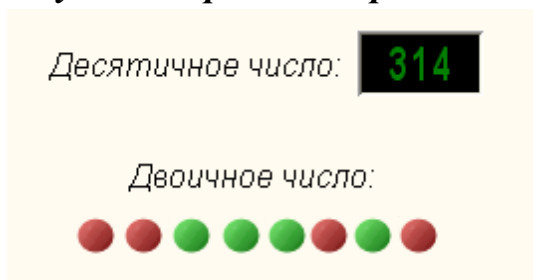
### *Проект в SCADA ZETView*





В этой схеме компонент [Битовая маска](#)<sup>[390]</sup> производит операцию маскирование согласно установленным битам. [Таймер](#)<sup>[183]</sup> служит для подачи импульса, по которому произойдет применение маски. [Цифровой индикатор](#)<sup>[362]</sup> нужен для отображения маски в десятичной форме.

### Результат работы проекта



Данный пример непосредственно в ZETView



### Математическое описание

Битовая маска – набор битов, который при помощи поразрядной конъюнкции или дизъюнкции накладывается на набор битов, в котором требуется установить или сбросить некоторые разряды.

Например, в числе  $01101111_2$  требуется скинуть пятый бит. Для этого мы делаем маску  $11011111_2$  и с помощью поразрядной операцией И производим логическое умножение этих двоичных чисел.

Поразрядное И	01101111
	11011111
Результат	01001111

Если в том же самом числе,  $01101111_2$  требуется установить четвертый бит, то это производится при помощи поразрядной операцией ИЛИ. Для этого

мы создаем маску  $00010000_2$ , далее производим логическое сложение.

Поразрядное ИЛИ	01101111
	00010000
Результат	01111111

## 15.4. Двоичное число

**Двоичное число** - компонент. Предназначен для перевода десятичного целого числа в его двоичное представление.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Число - вывод, на который поступает десятичное число для перевода.

#### ➤ Выходные:

- Бит 1 - вывод, с которого поступает состояние младшего (0-го) бита переведенного числа.
- ...
- Бит N - вывод, с которого поступает состояния N-го бита переведенного числа.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

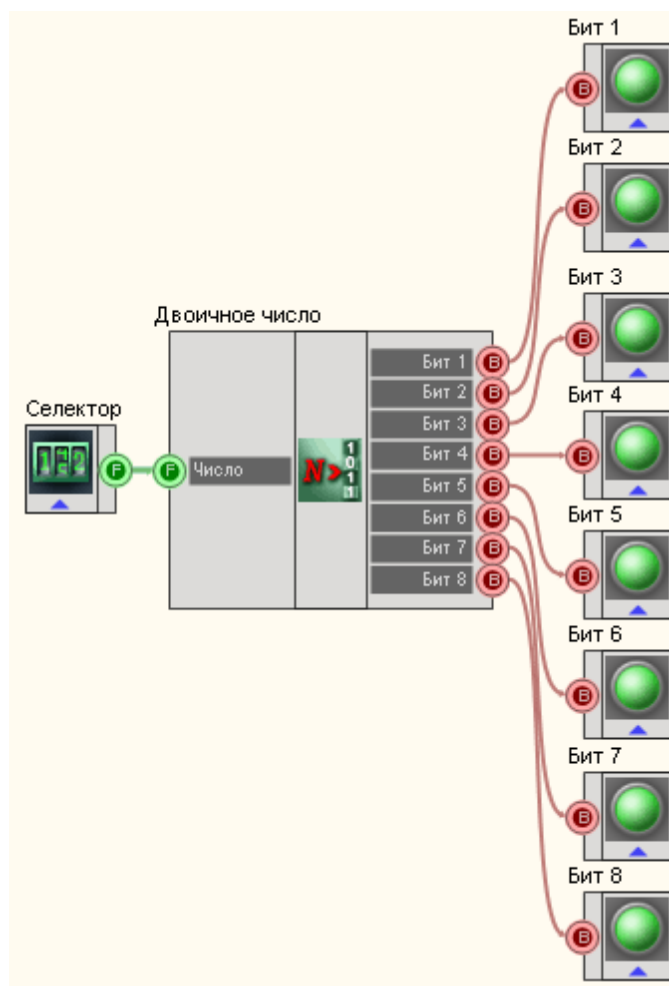
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- BitQuantity (1) - установка числа двоичных разрядов для перевода числа.



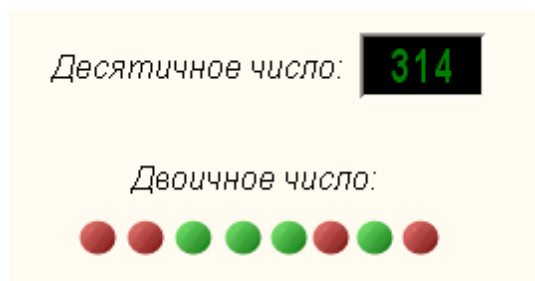
## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Двоичное число](#)<sup>[392]</sup> служит для перевода целого десятичного числа в двоичную форму. [Селектор](#)<sup>[213]</sup> нужен для ввода требуемого числа. [Световые индикаторы](#)<sup>[356]</sup> служат для графического отображения двоичной формы (зеленый цвет - логическая единица, красный - ноль).

### Результат работы проекта



Данный пример непосредственно в ZETView



### Математическое описание

Двоичная система исчисления - это позиционная система счисления с основанием 2. В данной системе число записывается только при помощи 0 и 1. Бит - один разряд двоичного кода (двоичная цифра). Двоичная система получила широкое распространение в электронике из-за ее простоты: смоделировать два состояния электронной схемы и затем их безошибочно различить неизмеримо проще, чем три, четыре и более. Восемь бит образуют байт. Это сложилось исторически, да и так удобнее: число кратно степени двойки, т.е. легко масштабируется. Шестнадцатиразрядное число – это просто два байта, тетрада (4 бита) – полубайт. Существует также двоично-десятичный код (binary-coded decimal, BCD-код). Они широко применяются в АЦП. Каждая значащая десятичная цифра в таком коде представляется четырьмя двоичными знаками и содержит десять значений сигнала от 0 до 9. Так например  $137_{10}$  – это  $0001\ 0011\ 0111_{BCD}$ . Заметим, что такое представление не эквивалентно  $137_{10} = 10001001_2$ . Очевидно, что двоично-десятичное представление не экономично с точки зрения использования разрядов, т.к. каждая группа из четырех бит способна “вместить” 16 состояний, а используется только 10 из них.

Преимущества BCD:

- упрощенный ввод-вывод чисел на индикацию;
- для дробных чисел (как с фиксированной, так и с плавающей запятой) при переводе в человекочитаемый десятичный формат и наоборот не теряется точность;
- упрощены умножение и деление на 10, а также округление.

Недостатки BCD:

- усложнены арифметические операции;
- требуется больше памяти.

Под цифровой электроникой мы подразумеваем схемы, в которых в каждой точке, как правило, можно определить только два состояния. Отсюда и такое внимание к двоичной системе исчисления. Например, транзистор открыт или закрыт. Как правило, выбирают в качестве параметра напряжение, а не ток. Отсюда и выбор уровней: Высокий и Низкий. Например для высокоскоростной КМОП логики входные напряжения от уровня земли до 1,5 В представляют как Низкий уровень, а напряжение в пределах 1,5 В до 5 В питания – как Высокий уровень. Эти состояния могут быть представлены различными битами (binary digits – двоичные разряды) информации. Например, замкнут ключ или разомкнут, есть или нет сигнала, уровень аналогового сигнала выше или ниже уровня – все это можно представить одним битом числа. Состояния Высокого и Низкого уровней определяют некоторым заданным образом “истинные” и “ложные” значения в булевой алгебре.

Большинство логических микросхем принадлежат к одной из разновидностей – ТТЛ(ТТЛ, Транзисторно-транзисторная логика) и КМОП (CMOS, Комплиментарные [транзисторы типа] металл-окисел-полупроводник). В технологии КМОП используются полевые транзисторы с изолированным затвором с каналами разной проводимости. Отличительной особенностью схем КМОП по сравнению с биполярными технологиями (ТТЛ, ЭСЛ и др.) является:

- очень малое энергопотребление в статическом режиме (в большинстве случаев можно считать, что энергия потребляется только во время переключения состояний);
- напряжение логической единицы практически равно напряжению питания, а напряжение логического нуля практически равно потенциалу “земли” (при ненагруженных выводах);
- порог переключения практически равен половине напряжения питания;
- широкий диапазон рабочих напряжений.

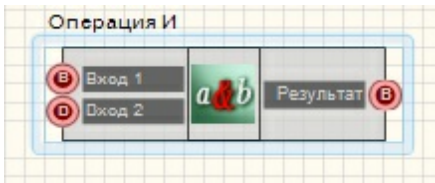
Но по сравнению с ТТЛ КМОП имеет низкое быстродействие, обусловленное тем, что изолированный затвор МОП-транзистора представляет

собой конденсатор довольно большой емкости. В совокупности с выходным резистивным сопротивлением, такой конденсатор образует фильтр нижних частот. В цифровой электронике важны не просто частотные свойства, а время задержки распространения сигнала на один логический элемент. Задержка возникает по причине того, что фронт сигнала не строго вертикальный, а наклонный. В связи с этим, когда сигнал на выходе начнет только нарастать (уменьшаться), в то время как на входе уже достигнет уже величины достаточной для переключения уровня. Другим следствием наличия высокой входной емкости у КМОП является то, что при переключении возникает импульс тока перезарядки этой емкости. Вследствие этого появляются затянутые фронты, отсюда элемент находится довольно длительное время в активном режиме, когда выходные транзисторы приоткрыты. В сочетании с высокоомными входами это все приводит к снижению помехозащищенности при переключениях.

## 15.5. Логическое И

**Логическое И** - компонент. Служит для реализации такой логической операции, как конъюнкция.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входы:

- Вход 1 - вывод, к которому подключается первый канал с бинарными значениями.
- Вход 2 - вывод, к которому подключается второй канал с бинарными значениями.

➤ Выходные:

- Выход - вывод, на котором выставляется значение булевой функции Логическое И от всех входных аргументов.

**Настраиваемые свойства:**

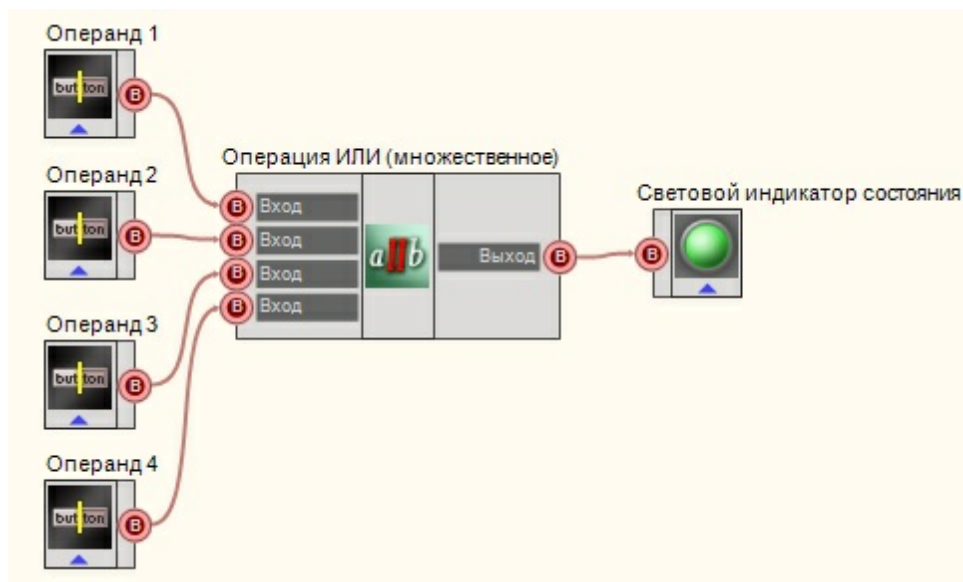
➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

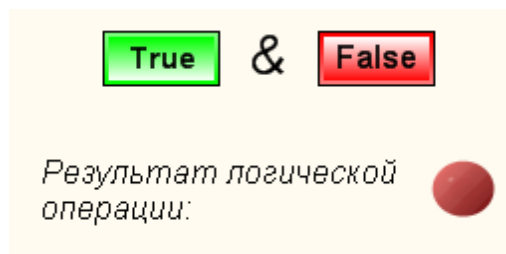
- FirstReaction (true) - устанавливается реакция на изменение данных на входе № 1
- SecondReaction (true) - устанавливается реакция на изменение данных на входе № 2

2

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Операция И](#)<sup>[396]</sup> служит для демонстрации работы булевой функции Логическое И. [Кнопки с фиксацией](#)<sup>[366]</sup> нужны для ввода бинарных значений. [Световой индикатор](#)<sup>[356]</sup> служат для графического отображения результата работы булевой функции Логическое И.

**Результат работы проекта**



Данный пример непосредственно в ZETView



### Математическое описание

Логический элемент И (конъюнкция, логическое умножение).

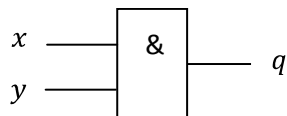


Таблица истинности элемента.

x	y	q
0	0	0
0	1	0
1	0	0
1	1	1

Логический элемент, реализующий функцию конъюнкции, называется схемой совпадения. Мнемоническое правило для конъюнкции с любым количеством входов звучит так: На выходе будет:

- 1 тогда и только тогда, когда на всех входах действуют «1»,
- 0 тогда и только тогда, когда хотя бы на одном входе действует «0»



## 15.6. Логическое И (множественное)

**Логическое И (множественное)** - компонент. Служит для реализации такой логической операции, как конъюнкция. Компонент принимает от двух и более входных логических значений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - вывод, к которому подключается первый канал с бинарными значениями.
- Вход 2 - вывод, к которому подключается второй канал с бинарными значениями.

#### ➤ Выходные:

- Выход - вывод, на котором выставляется значение булевой функции Логическое И от входных аргументов.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

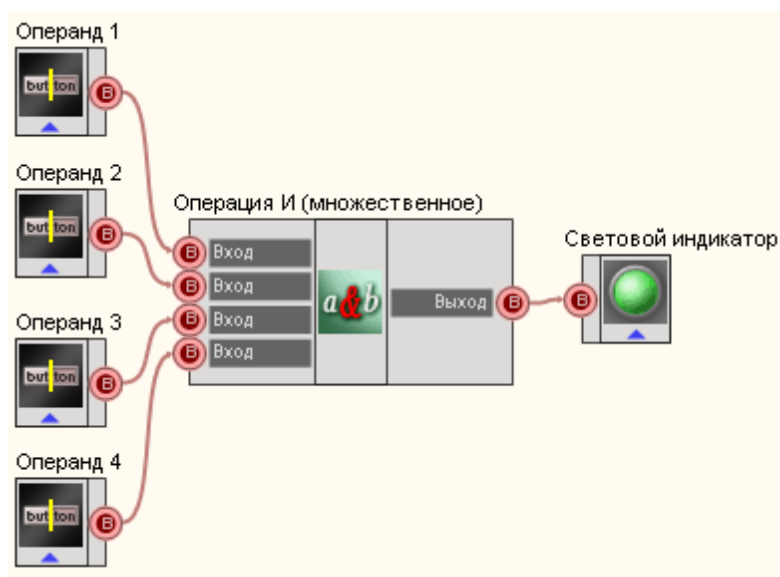
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Input (2) - установка количества входов.



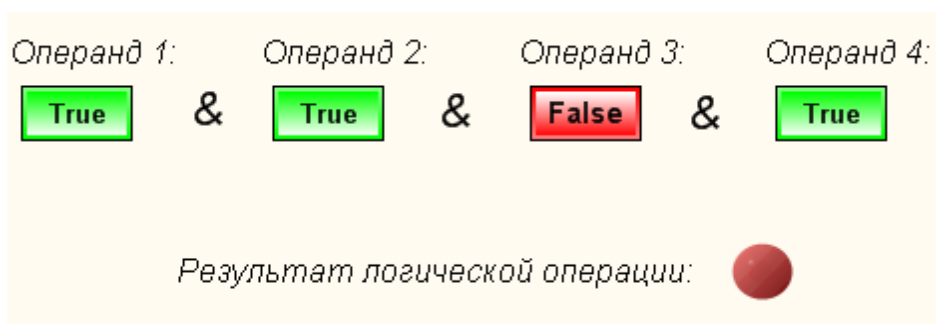
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Операция И \(множественное\)](#)<sup>[399]</sup> служит для реализации функции логического умножения. [Кнопки с фиксацией](#)<sup>[366]</sup> нужны для ввода бинарных значений-состояний. [Световой индикатор](#)<sup>[356]</sup> служит для графического отображения результата работы булевой функции Логическое И (для множества аргументов).

### *Результат работы проекта*

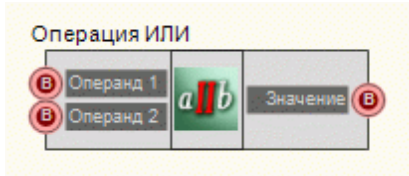


Данный пример непосредственно в ZETView

## 15.7. Логическое ИЛИ

**Операция ИЛИ** - компонент. Служит для реализации такой логической операции, как дизъюнкция.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Операнд 1 - канал №1 является первым аргументом.
- Операнд 2 - канал №2 является вторым аргументом.

## ➤ Выходные

- Значение - выходной канал, на котором формируются результат логического сложения двух операнд.

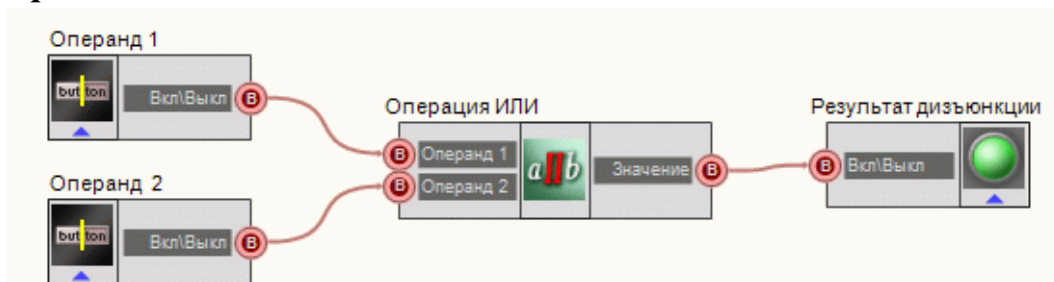
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - устанавливается реакция на изменение данных на входе № 1
- SecondReaction (true) - устанавливается реакция на изменение данных на входе № 2

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Логическое ИЛИ](#)<sup>[400]</sup> производит операцию дизъюнкции с двумя входящими значениями. [Кнопки с фиксацией](#)<sup>[366]</sup> служат для ввода численных значений самих переменных. Результат дизъюнкции выводится при помощи [Светового индикатора](#)<sup>[356]</sup>.

### Результат работы проекта

Операнд 1: True    OR    False    Операнд 2:

Результат  
логической операции: ●

Данный пример непосредственно в ZETView



### Математическое описание

Логический элемент ИЛИ (логическое сложение, дизъюнкция).

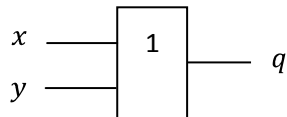


Таблица истинности элемента.

x	y	q
0	0	0
0	1	1
1	0	1
1	1	1

Мнемоническое правило для дизъюнкции с любым количеством входов звучит так: На выходе будет:

- 1 тогда и только тогда, когда хотя бы на одном входе действует «1»,
- 0 тогда и только тогда, когда на всех входах действуют «0»

## 15.8. Логическое ИЛИ (множественное)

**Логическое ИЛИ (множественное)** - компонент. Служит для реализации такой логической операции, как дизъюнкция. Компонент принимает от двух и более входных логических значений.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход 1 - вывод, к которому подключается первый канал с бинарными значениями.
- Вход 2 - вывод, к которому подключается второй канал с бинарными значениями.

#### ➤ Выходные:

- Выход - вывод, на котором выставляется значение булевой функции Логическое ИЛИ от входных аргументов.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

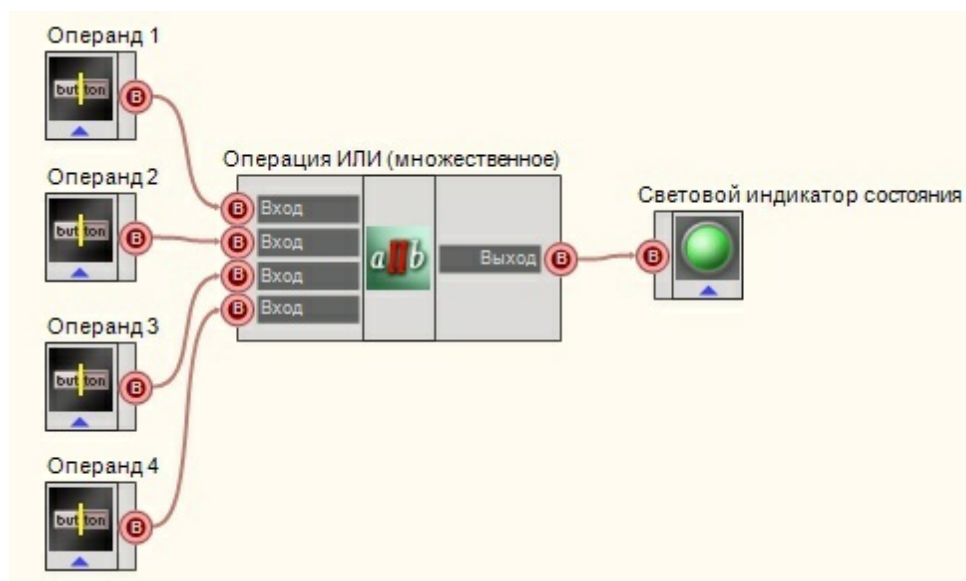
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Input (2) - установка количества входов.



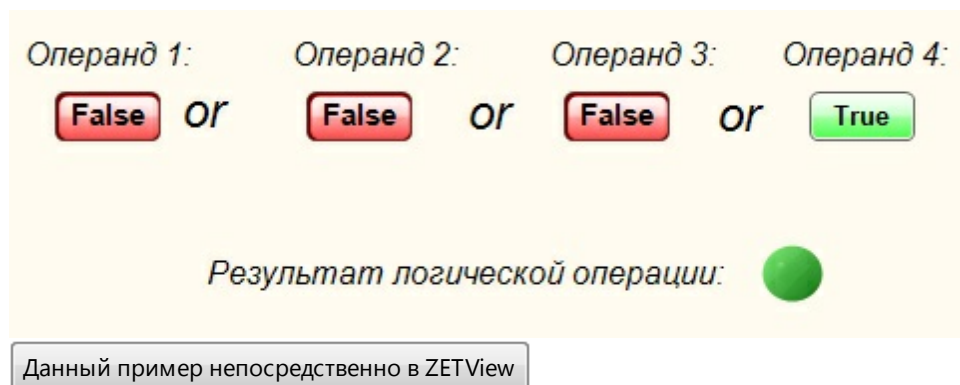
### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Операция ИЛИ \(множественное\)](#)<sup>[399]</sup> служит для реализации функции логического умножения. [Кнопки с фиксацией](#)<sup>[366]</sup> нужны для ввода бинарных значений-состояний. [Световой индикатор](#)<sup>[356]</sup> служит для графического отображения результата работы булевой функции Логическое ИЛИ (для множества аргументов).

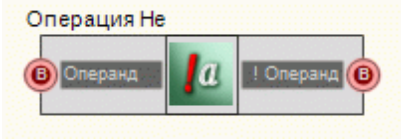
### *Результат работы проекта*



## 15.9. Логическое Не

**Логическое Не** - компонент. Предназначен для реализации операции логического отрицания.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Операнд - вывод, на который поступает бинарные значения, которые требуется инвертировать.

## ➤ Выходные:

- ! Операнд - вывод, с которого поступают инвертированные значения.

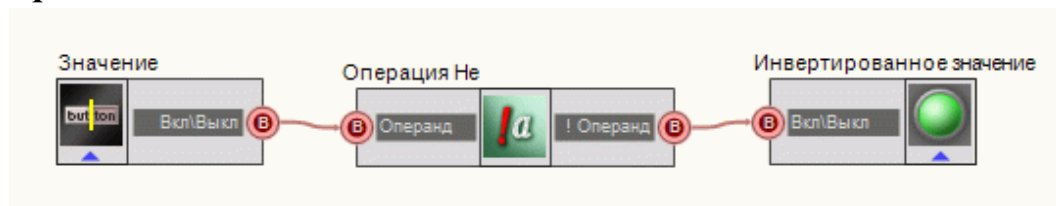
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

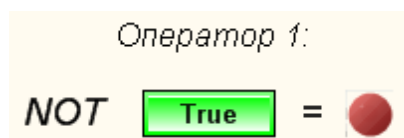
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства:

Не имеет.

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Операция Не](#)<sup>[404]</sup> служит для инвертирования дискретного значения, поступающего с [Кнопки с фиксацией](#)<sup>[366]</sup>. [Световой индикатор](#)<sup>[356]</sup> нужен для графического отображения логического уровня на выходе.

**Результат работы проекта**

Данный пример непосредственно в ZETView



### Математическое описание

Логический элемент Не инвертирует входной сигнал. Инверсию обозначает кружок на выходе схемы.

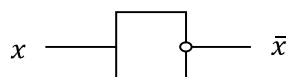


Таблица истинности элемента

x	НЕ
0	1
1	0

Мнемоническое правило для отрицания звучит так: На выходе будет:


- 1 тогда и только тогда, когда на входе «0»
- 0 тогда и только тогда, когда на входе «1»

## 15.10. Наложение маски

**Наложение маски** - компонент. Служит для маскирования — установки, сброса или проверки отдельных битов (позволяет осуществлять управление устройствами, имеющих 2 состояния).

**Внешний вид компонента:**



Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Число - контакт, на который подается само число, с которым будет проводиться операция маскирования.
- Маска - контакт, на который подается маска, которая при помощи поразрядной конъюнкции будет наложена на требуемое число.

#### ➤ Выходные

- Выход - контакт, на котором выставляется низкий логический уровень в том случае, когда все биты маски нулевые, и высокий, когда значение маски отлично от нуля.
- Выход - контакт, с которого выводится значения числа, после проведения над ним операции маскирования.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

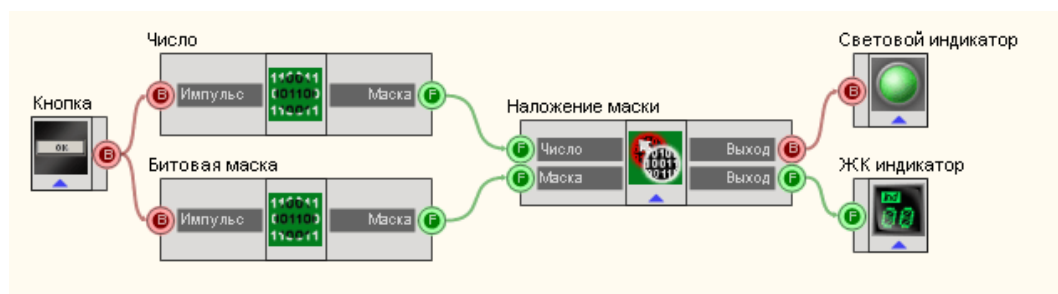
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - реакция на изменение маскируемого числа.
- SecondReaction (true) - реакция на изменение маски.
- Value (0) - значение по умолчанию числа.
- Mask (0) - значение по умолчанию маски.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Наложение маски](#)<sup>[406]</sup> производит операцию маскирования. [Битовые маски](#)<sup>[390]</sup> используются для двоичного представления Числа и Маски. По нажатию на [Кнопку без фиксации](#)<sup>[367]</sup> происходит передача входных значений в [Наложение маски](#)<sup>[406]</sup>. [ЖК индикатор](#)<sup>[348]</sup> необходим для графического отображения в десятичной форме числа после операции маскирования. [Световой индикатор](#)<sup>[356]</sup> нужен для индикации состояния маски (все нули или хотя бы одна единица).


### Результат работы проекта

Число

Биты	1	2	3	4	5	6	7	8
Значение	1	1	0	1	0	1	1	1

Битовая маска

Биты	1	2	3	4	5	6	7	8
Значение	0	0	0	0	1	1	0	1

**Маскировать** 

Число после маскирования:

160.00

Данный пример непосредственно в ZETView

## Глава 16.Массивы

### 16.1.Детерминированный массив

**Детерминированный массив** - компонент. Предназначен для передачи в проект массива значений при поступлении синхроимпульса.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Синхро - контакт, при подаче на который импульса, идет считывание массива из файла или другого источника в память и последующая передача данных в проект.

##### ➤ Выходные:

- {Y} - контакт, с которого поступают численные значения массива Y.
- {X} - контакт, с которого поступают численные значения массива X.
- Синхро - контакт, на котором в зависимости от состояния считывания элементов массива из выбранного источника, выставляется 1 или 0 (1 - считывание данных, 0 - режим ожидания).

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

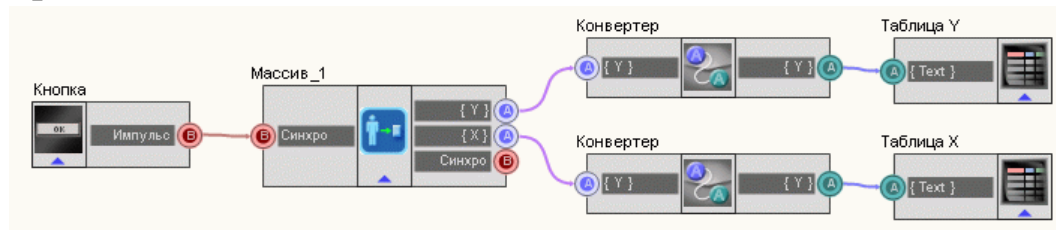
- Constant (0) - значение константы, из которой будет сформирован массив.
- K<sub>lin</sub> (0) - коэффициент k, в случае задания массива через формулу прямой (kx+b).
- B<sub>lin</sub> (0) - коэффициент b при задании массива через формулу прямой (kx+b).
- A<sub>sq</sub> (0) - коэффициент a при задании массива через формулу ax\*x+bx+c.
- B<sub>sq</sub> (0) - коэффициент b при задании массива через формулу ax\*x+bx+c.
- C<sub>sq</sub> (0) - коэффициент c при задании массива через формулу ax\*x+bx+c.
- FormulaQuantity (0) - количество значений расчета массива по формуле.
- FormulaStart (0) - начальное значение x при расчете массива по формуле.
- FormulaFinish (0) - конечное значение x при расчете массива по формуле.
- Formulatype (Константа) - тип формулы:

1. Константа.
  2. Линейная.
  3. Квадратичная
- FileName - выбор файла, в котором хранится сам массив.
  - formtype (Таблица) - источник детерминированного массива:
    1. Таблица - ввод вручную значений массива.
    2. Файл - получение значений массива из файла.
    3. Формула - получение значений массива при помощи формул линейной, квадратичной зависимости и константы.
  - ColNumber (1) - номер столбца в файле данных для считывания массива
  - TableSize (0) - размер таблицы
  - readchannel (Канал 1) - номер канала при считывании wav -файла
  - Kexp (1) - Коэффициент k при задании массива через формулу  $\exp(kx)$ .



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Детерминированный массив](#)<sup>[409]</sup> предназначен для формирования массива значений, рассчитанных по формуле линейной зависимости. [Конвертеры \(числовой массив - строковый массив\)](#)<sup>[378]</sup> нужен для преобразования типа данных в форму, совместимую с [Таблицами данных](#)<sup>[496]</sup> (этот компонент позволяет отображать данные массива в форме таблицы). [Кнопка без фиксации](#)<sup>[367]</sup> нужна для подачи синхроимпульса, по которому будет совершена передача элементов из Детерминированного массива на таблицы.

### Результат работы проекта

Таблица значений X	Таблица значений Y
0	3
0.714286	6.571429
1.428571	10.142858
2.142857	13.714285
2.857143	17.285715
3.571429	20.857143
4.285714	24.42857
5	28
5.714286	31.57143
6.428572	35.14286
7.142858	38.714287
7.857143	42.285713
8.571428	45.85714
9.285714	49.42857
10	53

**Передача значений**

Данный пример непосредственно в ZETView

## 16.2. Детерминированный массив из архива

**Детерминированный массив из архива** - компонент. Предназначен для передачи массива значений из архива данных \*.ana, \*.anp в массив при поступлении синхроимпульса.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Директория - контакт, указывающий путь к папке, содержащий архив данных \*.ana, \*.anp. и местоположения на диске.
- Дата - контакт, содержащий время начала данных в формате ДД.ММ.ГГГГ.
- Время - контакт, указывающий время начала данных в формате ЧЧ.ММ.СС.
- Интервал - входной контакт интервала забора данных, с.
- Импульс - входной контакт для прочтения данных из архива.

## ➤ Выходные

- {Y} - контакт, с которого поступают численные значения массива (мгновенные значения сигнала) Y.
- {X} - контакт, с которого поступают численные значения массива (мгновенные значения сигнала) X.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

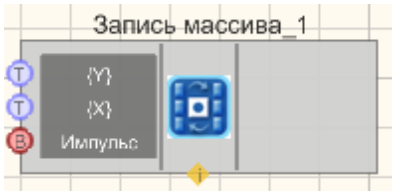
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Directory - путь к папке, содержащий архив данных \*.ana, \*.anp.
- Number (1) - номер канала в архиве.
- Date - дата начала данных в формате ДД.ММ.ГГГГ.
- Time - время начала данных в формате ЧЧ.ММ.СС.
- Interval (1) - интервал забора данных, с.

## 16.3. Запись массива

**Запись массива** - компонент. Предназначен для записи массивов данных в специальный файл на диске.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- {Y} - контакт, массив значения для записи.
- {X} - контакт, массив разметки по оси X для записи.
- Импульс - контакт импульса на запись массива в файл.

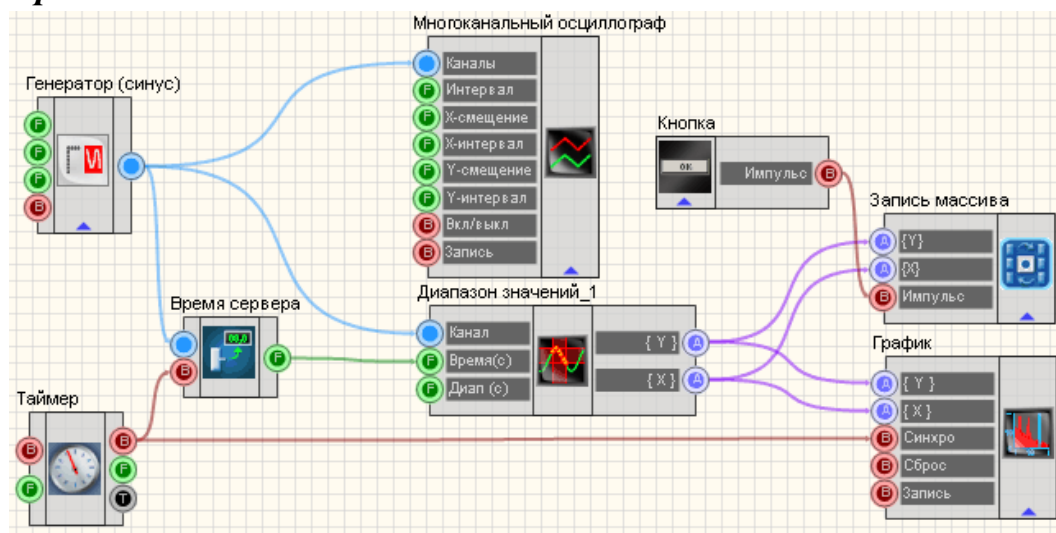
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

## ➤ Частные свойства:

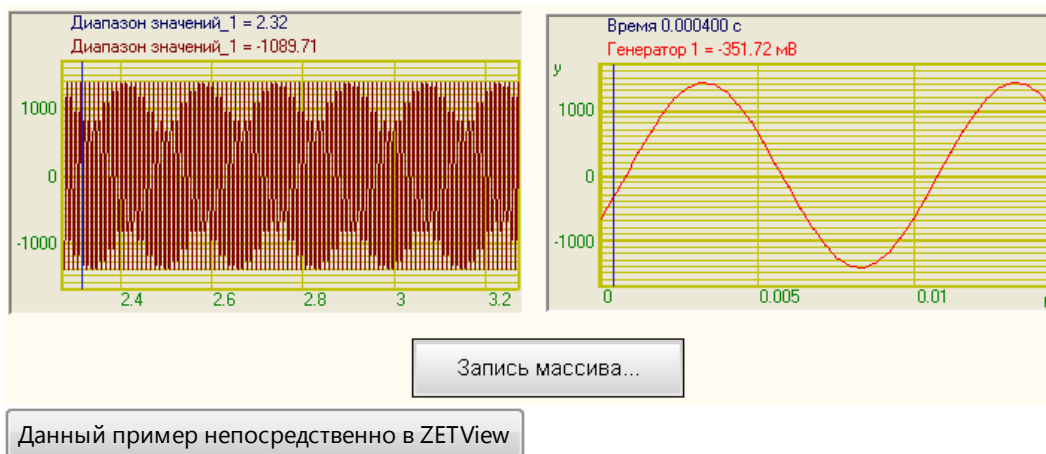
- FileName - установка названия файла записи и местоположения на диске.

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Диапазон значений](#) <sup>290</sup> предназначена для получения мгновенных значений сигнала канала в заданном диапазоне

времени. [Таймер](#)<sup>[183]</sup> служит для подачи синхронизирующих импульсов на [Время сервера](#)<sup>[180]</sup> и [График](#)<sup>[490]</sup>. Это нужно, чтобы [График](#)<sup>[490]</sup> и [Диапазон значений](#)<sup>[290]</sup> работали в едином временном пространстве. [Синусоидальный сигнал](#)<sup>[247]</sup> нужен для создания пробного сигнала, а [Многоканальный осциллограф](#)<sup>[494]</sup> - для его текущего графического отображения. [Запись массива](#)<sup>[412]</sup> позволяет реализовать запись сигнала в отдельный файл.

### Результат работы проекта



## 16.4.Матрица

**Матрица** - компонент. Предназначен для выдачи данных находящихся в строке/столбце, по запросу.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Номер - контакт, на который нужно подать номер строки/столбца, которые требуется вывести.

#### ➤ Выходные:

- {Данные} - выходной массив данных столбца/строки.



**Настраиваемые свойства:**

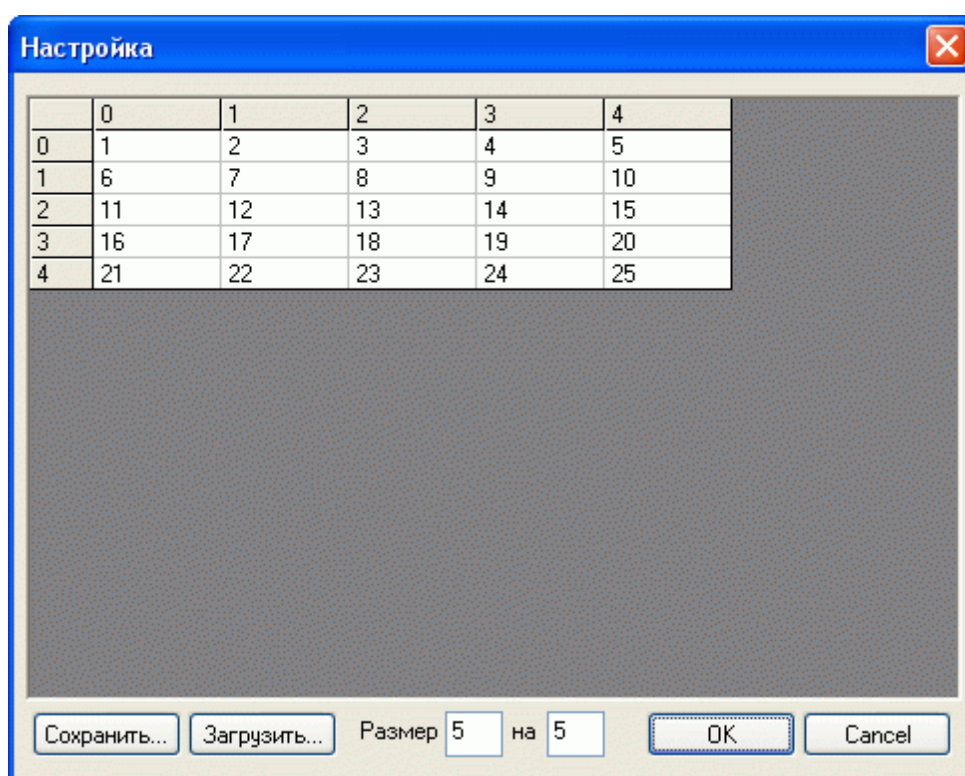
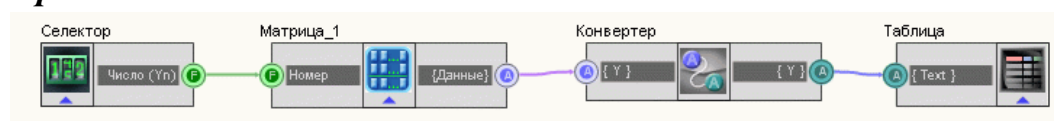
➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

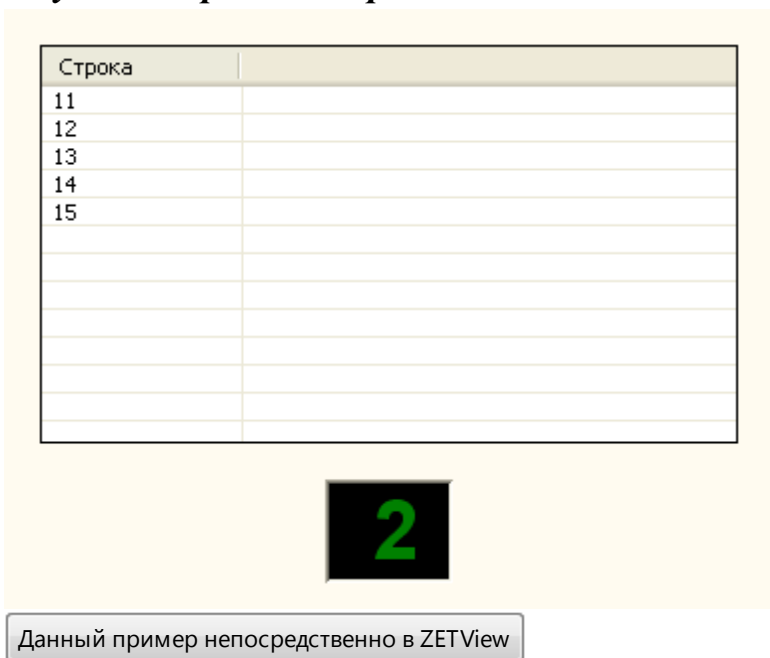
- outtype (Выдача строки по номеру) - способ выдачи матрицы:
  1. Выдача строки по номеру.
  2. Выдача столбца по номеру.
- Number (0) - номер выдаваемых строки/столбца.

При вызове контекстного меню компонента, необходимо выбрать Редактировать компонент, после чего откроется редактирования матрицы:

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Матрица](#)<sup>[414]</sup> предназначена для вывода по запросу значений, записанных в строке. [Конвертер](#)<sup>[378]</sup> служит для преобразования числового массива в строковый. [Селектор](#)<sup>[213]</sup> нужен для передачи на [Матрицу](#)<sup>[414]</sup> номера требуемой строки. [Таблица](#)<sup>[496]</sup> служит для отображения значений строки.

### Результат работы проекта



## 16.5.Формировщик массивов (дата-массив)

<Раздел находится в разработке>

**Формировщик массивов (дата-массив)** - компонент. Предназначен создания массива дат.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входы:

- Импульс - импульс для записи текущего значения времени-дата в массив.
  - Сброс - при подаче импульса на данный контакт происходит обнуление массива.
  - Размер - установка числа элементов в массиве.
- Выходные:
- Y - сформированный массив времени-дата.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ArrayAutoSize (true) - включение/выключение авторазмера массива.
- ArraySize (0) - установка размера массива.

## 16.6.Формировщик массивов (деление строки)

**Разделитель строки на подстроки** - компонент. Предназначен для разбиения одной строки на несколько строковых массивов.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входные:

- Вход - контакт, на который нужно подавать текст, который требуется разбить на части.

➤ Выходные:

- Массив - выходной строковый массив.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

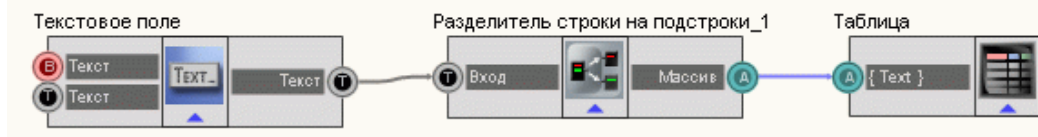
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Delimiter (;) - символ-разделитель.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Разделитель строки на подстроки<sup>\[417\]</sup>](#) предназначен для разбиения на части входного текста. [Таблица<sup>\[496\]</sup>](#) служит для графического отображения результата. [Текстовое поле<sup>\[561\]</sup>](#) нужно для организации ввода текста.

### Результат работы проекта




Данный пример непосредственно в ZETView

## 16.7.Формировщик массивов (массив-массив)

**Формировщик массивов (массив-массив)** - компонент. Формирует массив, производя одно из выбранных действий над входными массивами.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y1} - контакт, на который нужно подавать первый массив, с которым требуется произвести действия.
- {Y2} - контакт, на который нужно подавать второй массив, с которым требуется произвести действия.

#### ➤ Выходные:

- F{Y1; Y2} - выходной, сформированный массив.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FirstReaction (true) - включение/выключение реакция на изменение на входе №1 (при активированном режиме, по изменению на данном входе, происходит пересчет данных на выходе).
- SecondReaction (true) - включение/выключение реакция на изменение на входе №2 (при активированном режиме, по изменению на данном входе, происходит пересчет данных на выходе).
- formatype (Объединение) - тип преобразования, производимого с входящими массивами:
  1. Объединение - соединение двух массивов в один.
  2. Поэлементное сложение - нахождение суммы соответствующих элементов массива № 1 и № 2.
  3. Поэлементное вычитание - нахождение разности соответствующих элементов массива № 2 из № 1.
  4. Поэлементное умножение - нахождение произведения соответствующих элементов массива № 1 и № 2.
  5. Поэлементное деление - нахождение частного соответствующих элементов массива № 1 на массив № 2.
  6. Взаимная корреляция - нахождение корреляции между массивами.
  7. Поэлементное среднее - нахождение среднеарифметического значения соответствующих элементов массива № 1 и массива № 2.
  8. Выборка по индексам - использование дополнительно поддерживаемых индексных структур для получения данных.

9. Поэлементный максимум - выбор среди соответствующих значений массива № 1 и массива № 2 максимальных значений.

10. Поэлементный минимум - выбор среди соответствующих значений массива № 1 и массива № 2 минимальных значений.

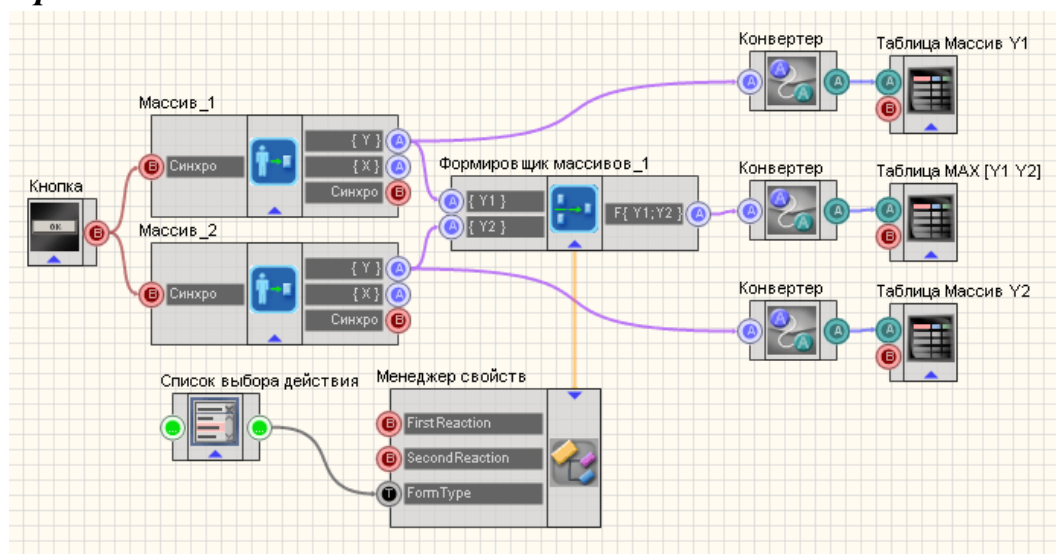
11. Поэлементный максимум (абсолютное значение)

12. Поэлементный минимум (абсолютное значение)



### Пример №1

#### Проект в SCADA ZETView



В этой схеме компонент [Формировщик массивов \(массив-массив\)](#)<sup>[418]</sup> предназначен для нахождения максимальных значений среди соответствующих элементов массива № 1 и массива № 2. [Конвертеры \(числовой массив - строковый массив\)](#)<sup>[378]</sup> нужен для преобразования типа данных в форму, совместимую с [Таблицами данных](#)<sup>[496]</sup> (этот компонент позволяет отображать данные массива в форме таблицы). [Детерминированные массивы](#)<sup>[409]</sup> нужны для создания массива № 1 и массива № 2 при помощи встроенного генератора значений.

#### Результат работы проекта

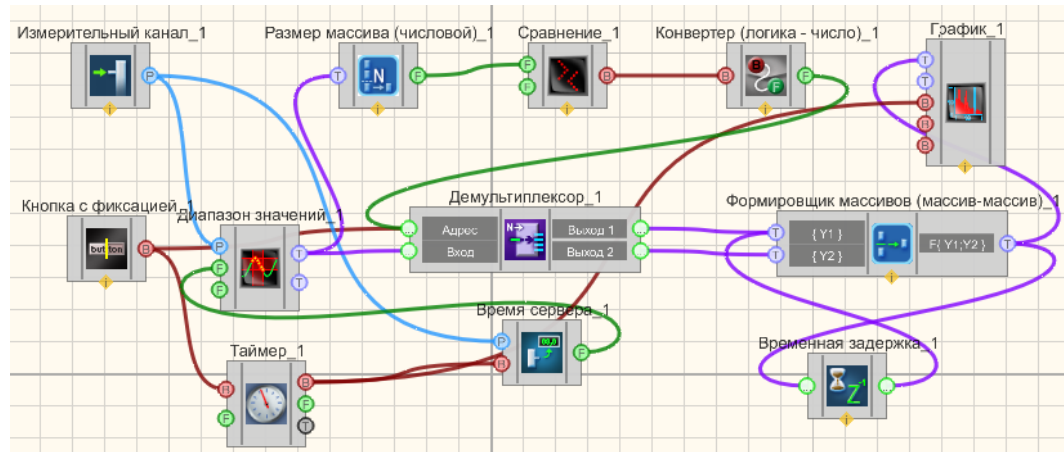
Массив Y1	Массив Y2	Результат
3	7	10
53	67	119
102	203	305
152	416	567
201	705	906
251	1071	1321
300	1513	1813
350	2032	2382
399	2627	3026
449	3299	3748
498	4048	4546
548	4873	5420
597	5774	6371
647	6752	7399
696	7807	8503

Данный пример непосредственно в ZETView



## Пример №2

### Проект в SCADA ZETView



### Результат работы проекта



Данный пример непосредственно в ZETView

## 16.8.Формировщик массивов (текст-массив)

**Формировщик массивов (текст-массив)** - компонент. Предназначен для заполнения массива символьными значениями.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Y n - контакт, на который нужно подавать символьные значения, которые требуется записать в массив.
- Сброс - при подаче импульса на данный контакт происходит обнуление массива.
- Размер - установка числа элементов в массиве.

➤ Выходные:

- Y - выходной сформированный массив.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

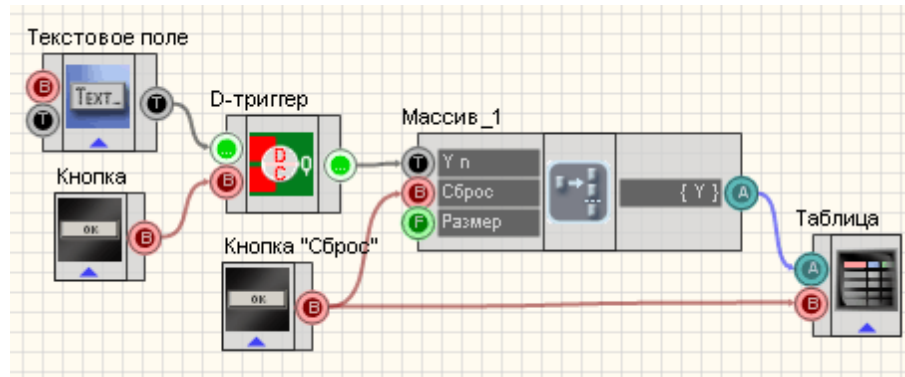


- Частные свойства (в скобках значение, установленное по умолчанию):
- ArrayAutoSize (true) - включение/выключение авторазмера массива.
  - ArraySize (0) - установка размера массива.



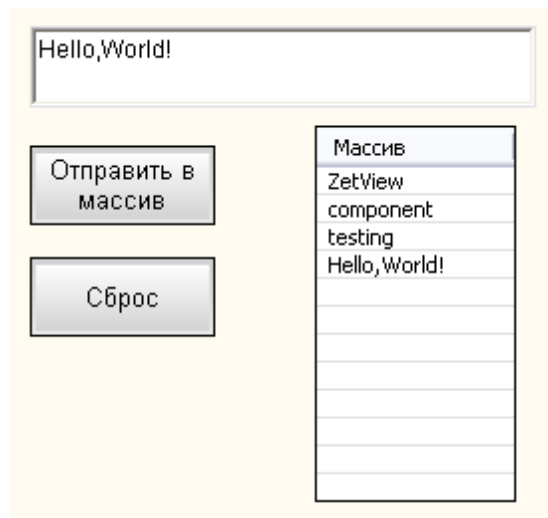
## Пример

### Проект в SCADA ZETView



В данной схеме компонент [Формировщик массивов \(текст-массив\)](#) [422] предназначен для вывода значений [Текстового поля](#) [561], при нажатии кнопки "Отправить в массив", подключенной к [D-триггеру](#) [384] данные с [текстового поля](#) [561] отправляются далее. [Таблица](#) [496] нужна для графического отображения массива строк.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 16.9.Формировщик массивов (число-массив)

**Формировщик массивов (число-массив)** - компонент. Предназначен для заполнения числового массива значениями.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входящие:

- Y n - контакт, на который нужно подавать числовые значения, которые требуется записать в массив.
- Сброс - при подаче импульса на данный контакт происходит обнуление массива.
- Размер - установка числа элементов в массиве.

➤ Выходные:

- Y - выходной числовой массив.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке](#).<sup>92</sup>

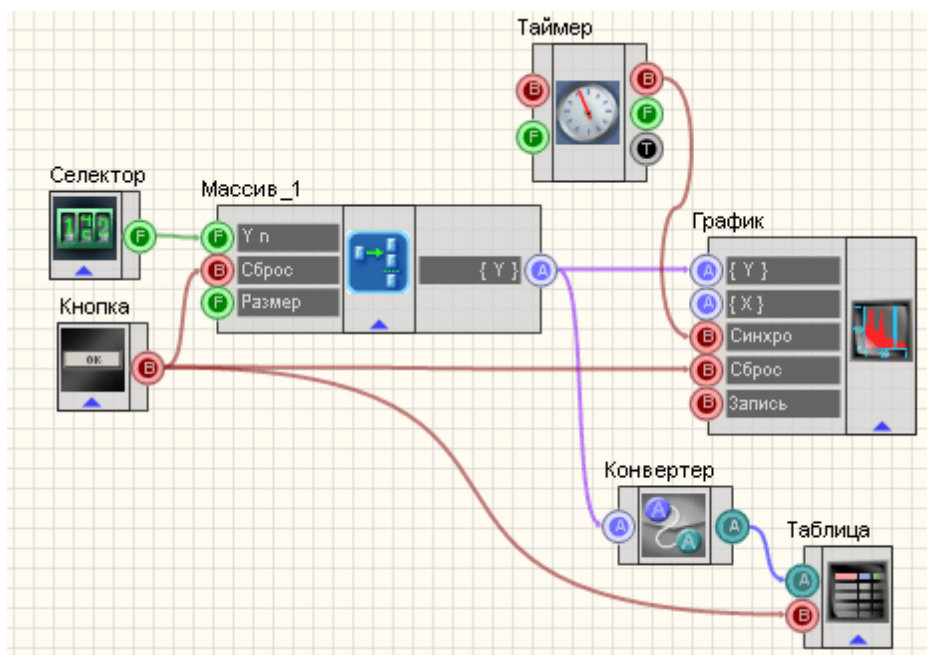
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ArrayAutoSize (true) - включение/выключение авторазмера массива.
- ArraySize (0) - установка размера массива.



*Пример*

*Проект в SCADA ZETView*



В этой схеме компонент [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup> предназначен для заполнения массива числами, которые поступают с [Селектора №1](#)<sup>[213]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> используется для обнуления массива. [Таймер](#)<sup>[183]</sup> служит для подачи синхроимпульсов на [График](#)<sup>[490]</sup>. [Конвертер \(числовой массив - строковый массив\)](#)<sup>[378]</sup> нужен для преобразования типа данных в форму, совместимую с [Таблицей данных](#)<sup>[496]</sup> (этот компонент позволяет отображать данные массива в форме таблицы).

### Результат работы проекта



Данный пример непосредственно в ZETView

## Глава 17.Массив - Значение

### 17.1.Значение элемента (строковый)

**Значение элемента (строковый)** - компонент. Предназначен для получения элемента массива по индексу.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- {Y} - контакт, на который нужно подавать значения строкового массива, который будет взят с заданным индексом.
- Индекс - контакт, на который поступает номер (индекс) элемента, который впоследствии будет передан на выход компонента.

##### ➤ Выходные:

- Число(Yn) - контакт, с которого поступает элемент, с заданным индексом.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>
- 

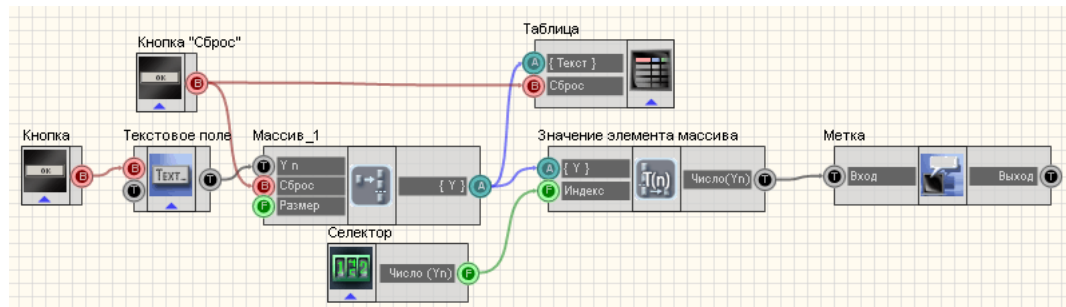
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IndexOfValue (0) - индекс требуемого элемента массива.



### Пример

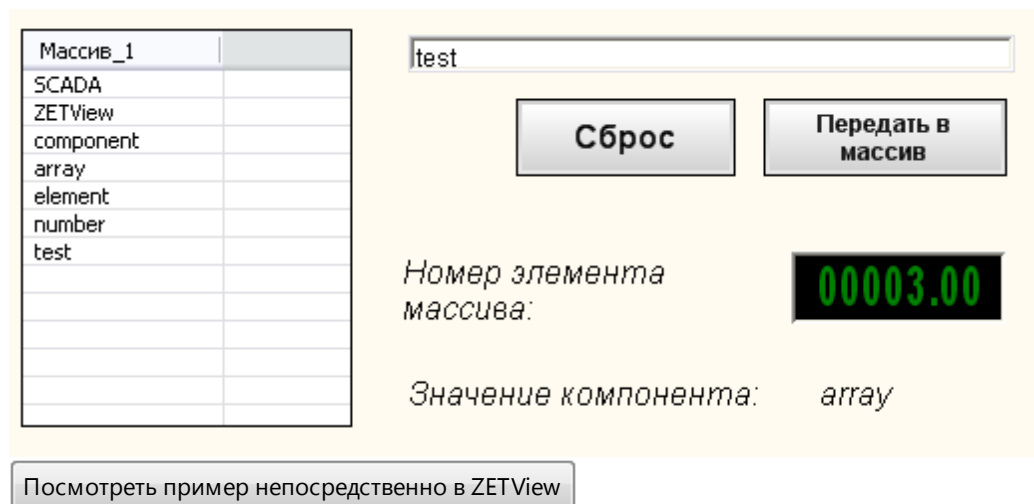
## Проект в SCADA ZETView



В данном проекте показана работа компонента ["Значение элемента \(строковый\)"](#)<sup>[426]</sup>.

Массив формируется пользователем, путем ввода значений в [текстовое поле](#)<sup>[561]</sup> (на компоненте выставлена настройка "передать по импульсу") и нажатия [кнопки без фиксации](#)<sup>[367]</sup> "Передать в массив". После этих действий строка отправляется на [Формировщик массивов \(текст-массив\)](#)<sup>[422]</sup> с которого поступает на [Таблицу данных](#)<sup>[496]</sup> (чтобы пользователь мог видеть, какие значения он вводит в массив). С помощью [селектора](#)<sup>[213]</sup> выставляется индекс нужного элемента массива. Компонент ["Значение элемента \(строковый\)"](#)<sup>[426]</sup> возвращает значение элемента по заданному индексу. Результат отображается помощью [метки](#)<sup>[554]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#)<sup>[422]</sup> и [Таблице данных](#)<sup>[496]</sup> сбрасывает значения массива и очищает таблицу.

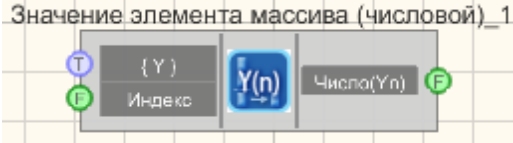
## Результат работы проекта



## 17.2.Значение элемента (числовой)

**Значение элемента (числовой)** - компонент. Предназначен для получения элемента массива по индексу.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- $\{Y\}$  - контакт, на который нужно подавать числовые значения массива, который будет взят с заданным индексом.
- Индекс - контакт, на который поступает номер (индекс) элемента, который впоследствии будет передан на выход компонента.

#### ➤ Выходные:

- Число( $Y_n$ ) - контакт, с которого поступает элемент, с заданным индексом.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

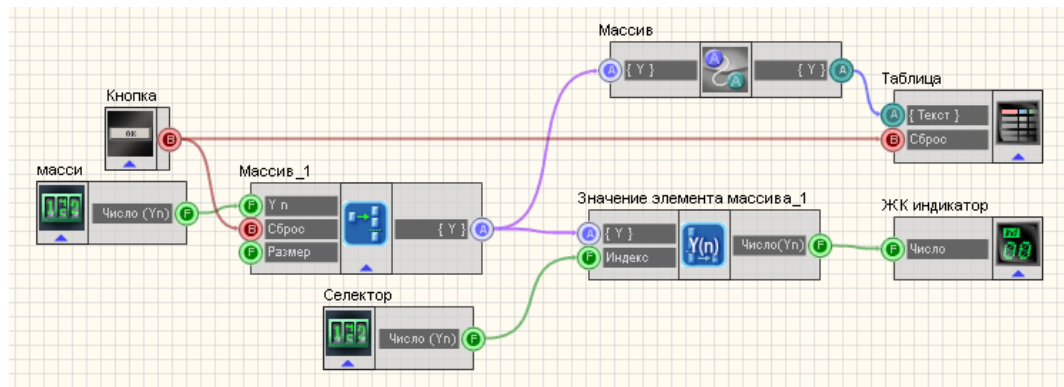
- IndexOfValue (0) - индекс требуемого элемента массива.
- FirstReaction (true) - реакция на изменение данных на входе №1
- SecondReaction (true) - реакция на изменение данных на входе №2



### *Пример*

#### *Пример 1*

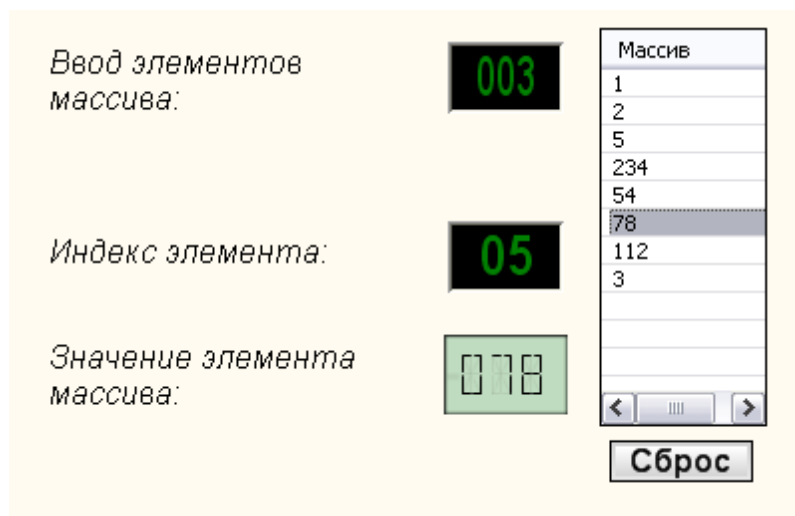
### *Проект в SCADA ZETView*



В данной схеме показана работа компонента [Значение элемента \(числовой\)](#) <sup>[428]</sup>.

С [селектора](#) <sup>[213]</sup> вводится массив на [Формировщик массивов \(число-массив\)](#) <sup>[424]</sup>, после чего данные поступают на [Конвертер \(числ.мас.-стр.мас.\)](#) <sup>[378]</sup>, где преобразуются из числового массива в строковый и выводятся в [Таблицу данных](#) <sup>[496]</sup>. Компонент ["Значение элемента массива"](#) <sup>[428]</sup> получает вводимый пользователем массив и индекс элемента поступающий со второго селектора и передает значение элемента, имеющего нужный индекс на [ЖК индикатор](#) <sup>[348]</sup>.

### Результат работы проекта



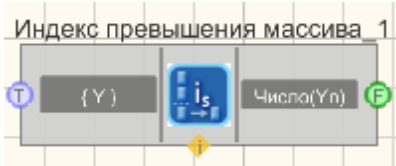
[Посмотреть пример непосредственно в ZETView](#)

[Пример 2 \(на примере компонента "Константа"\)](#) <sup>[205]</sup>

## 17.3. Индекс превышения

**Индекс превышения** - компонент. Предназначен для получения индекса числа, значение которого превышает заданное.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - контакт, на который нужно подавать значения массива.

#### ➤ Выходные:

- Число(Yn) - контакт, с которого поступает индекс элемента, значение которого превышает заданное.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

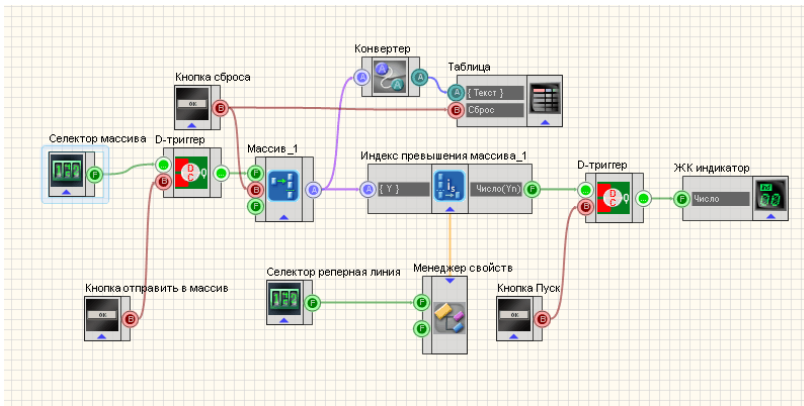
- RepetLev (0) - уровень реперной линии при поиске индекса превышения.



### *Пример*

### *Проект в SCADA ZETView*

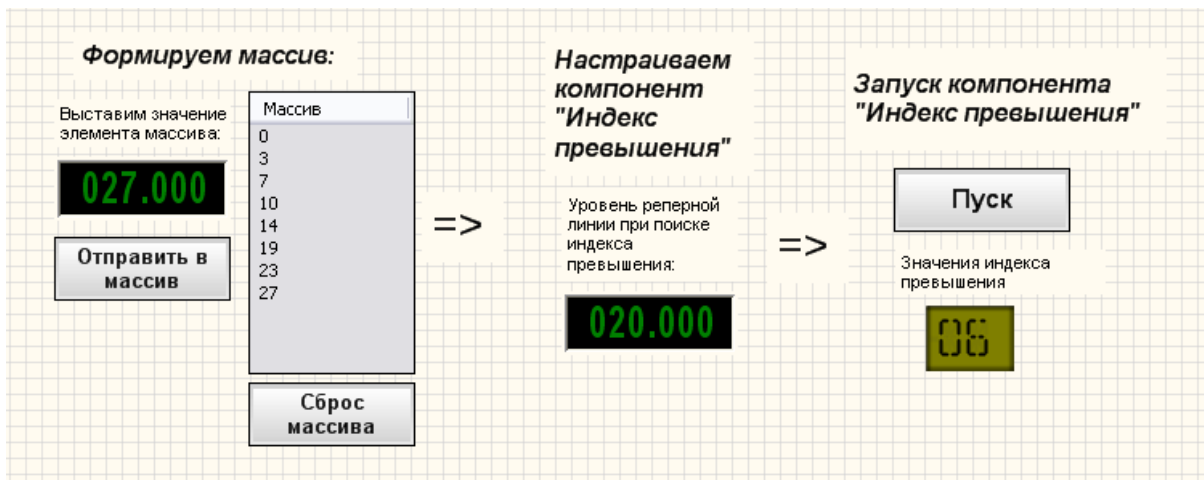




В данной схеме показана работа компонента [Индекс превышения](#)<sup>[430]</sup>. На вход {Y} компонента [Индекс превышения](#)<sup>[430]</sup> подается массив, сформированный компонентом [Формировщик массива](#)<sup>[424]</sup>. Элементы массива формируются с помощью [Селектора](#)<sup>[213]</sup>, на котором мы можем вручную выставить нужное нам число, затем число с селектора отправляется на [D-триггер](#)<sup>[384]</sup>. С помощью не фиксирующей кнопки Отправить в массив мы даем триггеру команду передать значение на [Селекторе](#)<sup>[213]</sup> [Формировщику массива](#)<sup>[424]</sup>. Кнопка сброса, также не фиксирующая кнопка, обнуляет [Формировщик массива](#)<sup>[424]</sup>, на случай если понадобится сформировать массив заново. Далее компонент [Индекс превышения](#)<sup>[430]</sup> складывает элементы полученного массива и передает его на выход Число(Yn). После этого число поступает на [Цифровой индикатор](#)<sup>[362]</sup>.

Для большей наглядности работы схемы к ней также подключены [Конвертер \(числ.массив - стр.массив\)](#)<sup>[378]</sup> и [Таблица данных](#)<sup>[496]</sup>, которые позволяют записывать вводимый нами массив в таблицу.

## Результат работы проекта

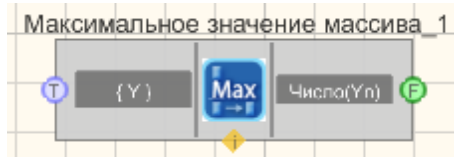


Посмотреть пример непосредственно в ZetView

## 17.4.Максимальное значение массива

**Максимальное значение массива** - компонент. Предназначен для получения максимального значения входного массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- {Y} - входной массив для поиска максимума.

➤ Выходные:

- ЧислоY(n) - полученное максимальное значение массива.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

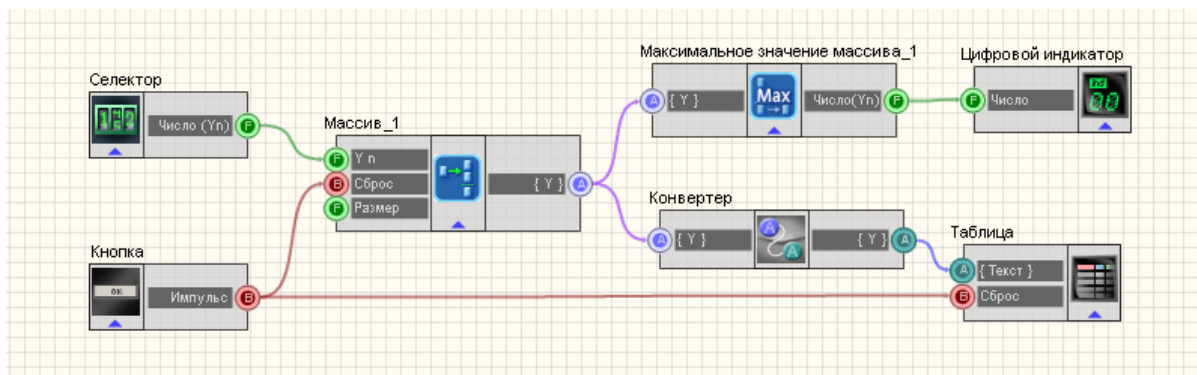
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Махtype (значение) - вариант искомой величины при поиске максимума в массиве (значение/индекс в массиве).



*Пример*

*Проект в Scada*

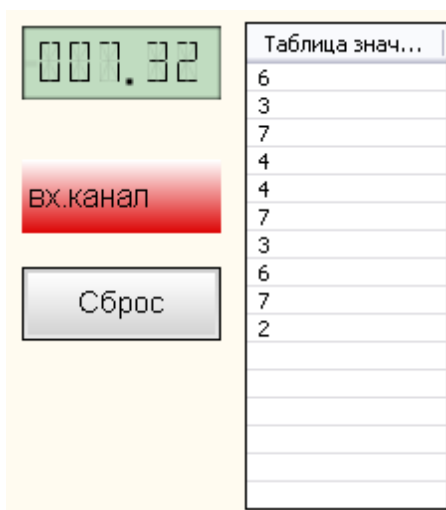


В данной схеме показана работа компонента Максимальное значение массива. На вход {Y} компонента Максимальное значение массива подается массив, сформированный компонентом [Формировщик массива](#)<sup>[424]</sup> с параметром ArraySize=10 (Размер массива). К Формировщику массива подключен [Селектор](#)<sup>[213]</sup>, который позволяет вручную выставить значение элемента массива, и кнопка сброса, на случай если понадобится сформировать массив заново. Компонент Максимальное значение массива получив 10 значений элементов массива выбирает наибольшее и передает его на выход Число(Yn). После этого число поступает на [Цифровой индикатор](#)<sup>[348]</sup>.

Для большей наглядности работы схемы к ней также подключены [Конвертер \(число - строка\)](#)<sup>[379]</sup> и [Таблица данных](#)<sup>[496]</sup>, которые позволяют записывать вводимый нами массив в таблицу.

Для того, чтобы увидеть результат, необходимо, чтобы к компьютеру было подключено физическое ZET-устройство. В списке каналов нужно выбрать один из генераторов ZETLab.

### Результат работы проекта

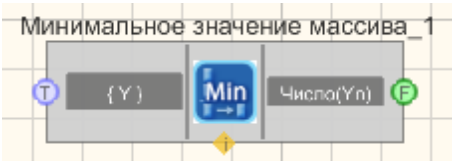


[Посмотреть пример непосредственно в ZetView](#)

## 17.5. Минимальное значение массива

**Минимальное значение массива** - компонент. Предназначен для получения минимального значения входного массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- {Y} - входной массив для поиска минимума.

➤ Выходные:

- ЧислоY(n) - полученное минимальное значение массива.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

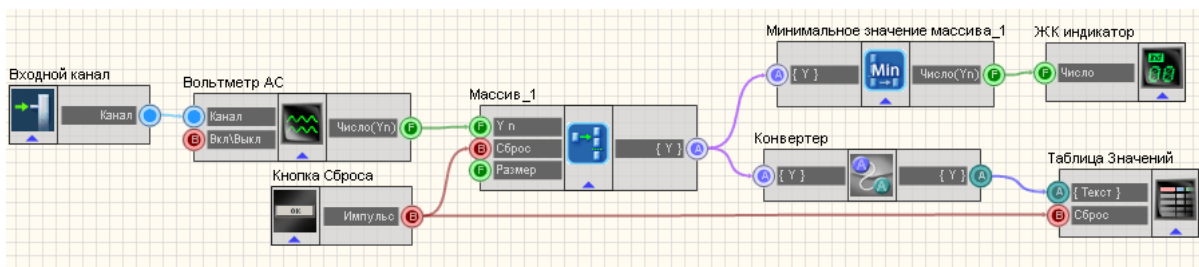
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Mintype (значение) - вариант искомой величины при поиске минимума в массиве (значение/индекс в массиве).



*Пример*

*Проект в Scada*

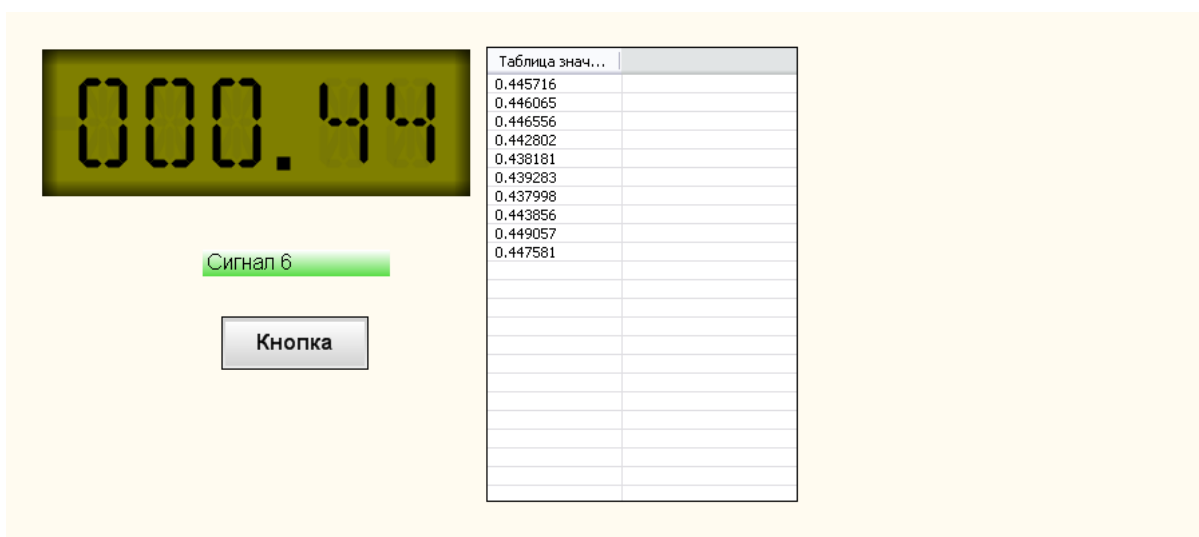


В данной схеме показана работа компонента Минимальное значение массива. На вход {Y} компонента Минимальное значение массива подается массив, сформированный компонентом [Формировщик массива](#)<sup>[424]</sup> с параметром ArraySize=10 (Размер массива). К Формировщику массива подключен [Входной канал](#)<sup>[227]</sup>, который передает сигнал с одного из каналов подключенного к компьютеру ZET-устройства. Кнопка сброса, на случай если понадобится сформировать массив заново. Компонент Минимальное значение массива получив 10 значений элементов массива выбирает наименьшее и передает его на выход Число(Yn). После этого число поступает на [ЖК индикатор](#)<sup>[348]</sup>.

Для большей наглядности работы схемы к ней также подключены [Конвертер \(число - строка\)](#)<sup>[379]</sup> и [Таблица данных](#)<sup>[496]</sup>, которые позволяют записывать вводимый нами массив в таблицу.

Для того, чтобы увидеть результат, необходимо, чтобы к компьютеру было подключено физическое ZET-устройство. В списке каналов нужно выбрать один из генераторов ZETLab.

## Результат работы проекта

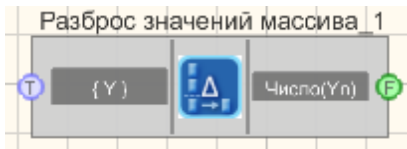


Посмотреть пример непосредственно в ZetView

## 17.6.Разброс значений

**Разброс значений** - компонент. Предназначен для отображения разницы значений между максимальным и минимальным элементами массива.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - входной числовой массив

#### ➤ Выходные

- Число(Yn) - разброс значений

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

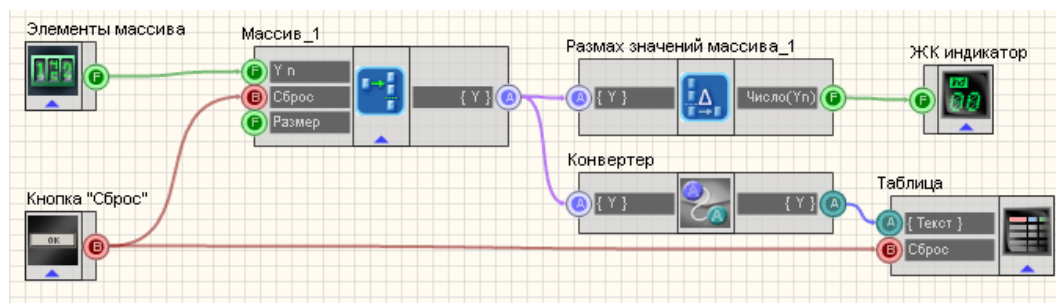
#### ➤ Частные свойства:

Не имеет.



### Пример

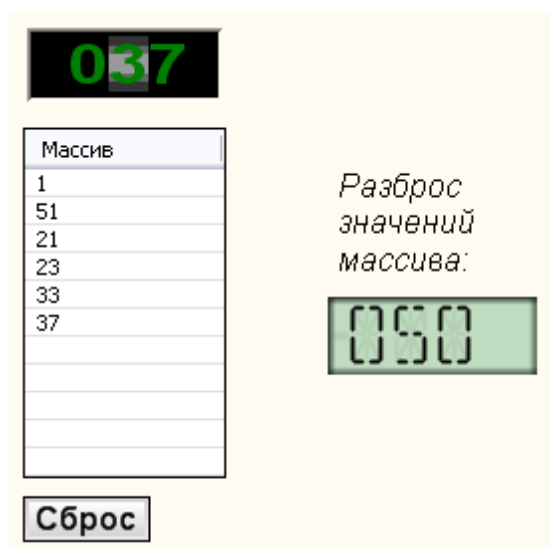
### Проект в SCADA ZETView



В данном проекте рассматривается работа компонента "Разброс значение".

Массив формируется с помощью компонента [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup> путем ввода значений на [селекторе](#)<sup>[213]</sup>, [кнопка без фиксации](#)<sup>[367]</sup> "Сброс" обнуляет массив и значения в [таблице](#)<sup>[496]</sup>. [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup> отправляет полученный массив компоненту "Разброс значений", а тот, в свою очередь вычисляет необходимую величину и отправляет результат на [ЖК индикатор](#)<sup>[348]</sup>.

### Результат работы проекта



Посмотреть пример непосредственно в ZETView

## 17.7.Размер массива (строковый)

**Размер строкового массива** - компонент. Предназначен для получения размера строкового массива. Выходные данные - количество элементов в строковом массиве.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входящие:

- $\{Y\}$  - входной строковый массив.

#### ➤ Выходные:

- Число $Y(n)$  - количество элементов в строковом массиве.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

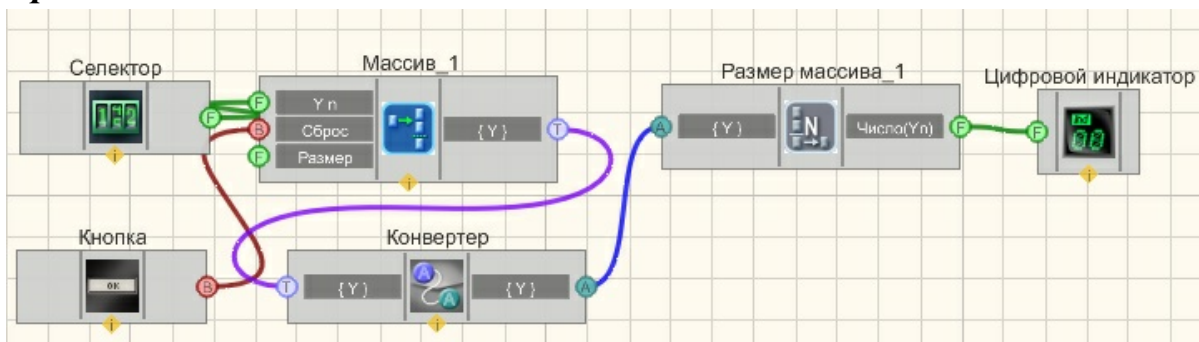
#### ➤ Частные свойства:

Не имеет.

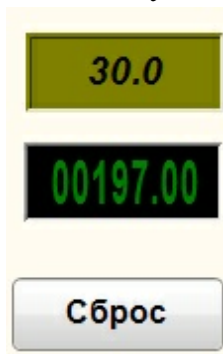


### Пример

### Проект в SCADA ZETView



### Результат работы проекта



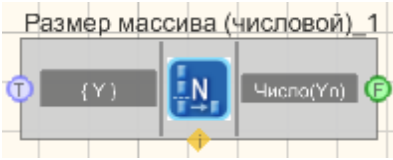


Посмотреть пример непосредственно в ZetView

## 17.8.Размер массива (числовой)

**Размер числового массива** - компонент. Предназначен для получения размера числового массива. Выходные данные - количество элементов в числовом массиве.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - входной числовой массив.

#### ➤ Выходные:

- ЧислоY(n) - количество элементов в числовом массиве.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

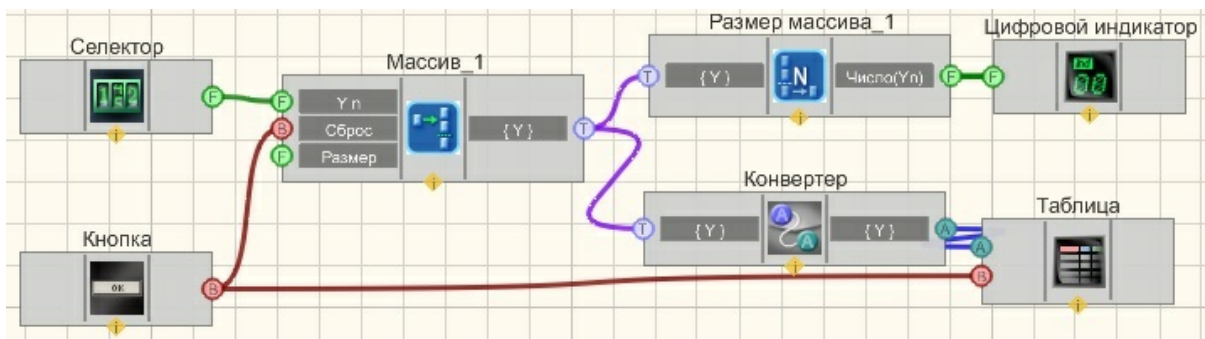
#### ➤ Частные свойства:

Не имеет.



### *Пример*

### *Проект в SCADA ZETView*



### Результат работы проекта

Таблица данн...	
784	
783	
782	
781	
780	
779	
778	

[Посмотреть пример непосредственно в ZetView](#)

## 17.9. Синхронизация массива

<Раздел находится в разработке>

**Синхронизация массива** - компонент.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- {Y} - входной числовой массив

## ➤ Выходные

- Число(Yn) - выходное значение массива.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

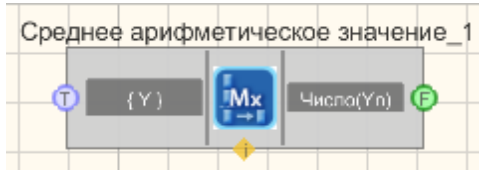
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- synchrotype (по нарастающему фронту) - тип синхронизации при обработке массива с помощью синхронизации.
  1. по нарастающему фронту.
  2. по ниспадающему фронту.
- SynchroLev (0) - уровень синхронизации при обработке массива с помощью синхронизации.
- SynchroWid (0) - ширина уровня синхронизации при обработке массива с помощью синхронизации.
- synchrometh (Ширина импульса) - вариант искомой величины при обработке массива с помощью синхронизации:
  1. Ширина импульса.
  2. Длительность фронта.
  3. Количество фронтов.

## 17.10. Среднее арифметическое значение

**Среднее арифметическое значение** - компонент. Вычисляет среднее арифметическое значение элементов входного массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- $\{Y\}$  - входной массив.

## ➤ Выходные:

- Число  $Y(n)$  - полученное среднее арифметическое значение элементов входного массива.

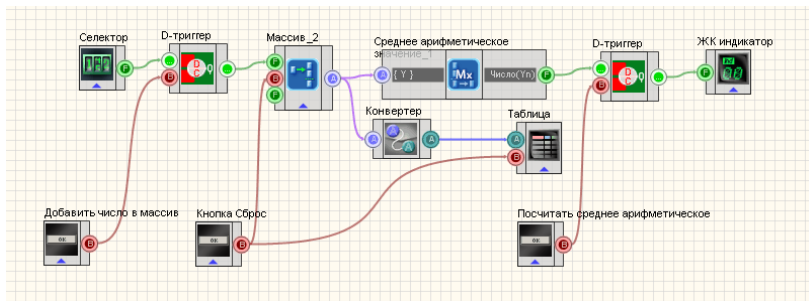
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

[Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства:

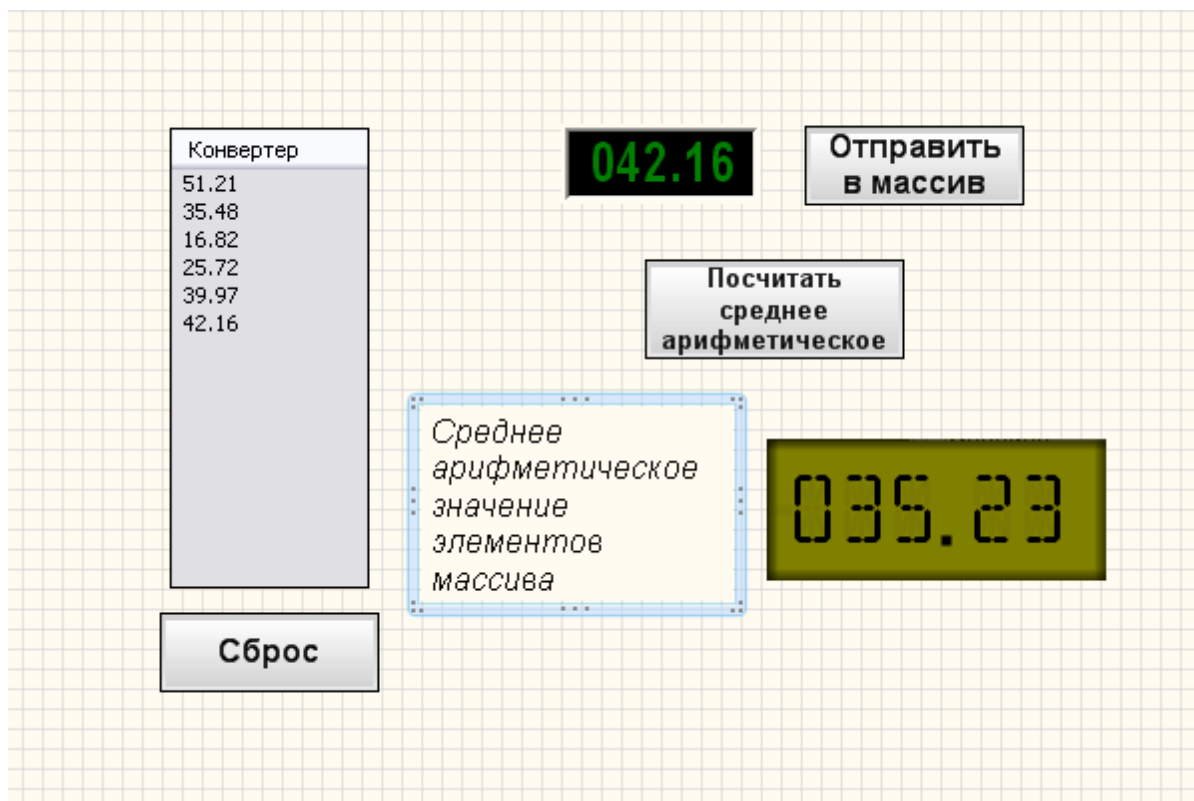
Не имеет.

**Пример****Проект в SCADA ZETView**

Данная схема описывает работу компонента [Среднее арифметическое значение](#)<sup>[441]</sup>. Массив формируется следующим образом:

на [Селекторе](#)<sup>[213]</sup> выставляется нужное нам значение и отправляется на [D-триггер](#)<sup>[384]</sup>; Кнопка без фиксации [Добавить число в массив](#) при нажатии отправляет [D-триггеру](#)<sup>[384]</sup> управляющий импульс, позволяющий передать значение, полученное с [Селектора](#)<sup>[213]</sup> [Формировщику массивов \(число-массив\)](#)<sup>[424]</sup>. Далее массив поступает на компонент [Среднее арифметическое значение](#)<sup>[441]</sup>, на выходе которого мы получаем число - среднее арифметическое, которое затем поступает на [ЖК индикатор](#)<sup>[348]</sup>. Компоненты [Таблица данных](#)<sup>[496]</sup> и [Конвертер \(числ.массив-стр.массив\)](#)<sup>[378]</sup> выводят массив в таблицу.

## Результат работы проекта



[Посмотреть пример непосредственно в ZetView](#)



### **Математическое описание**

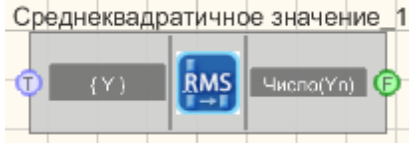
Среднее арифметическое набора чисел — это сумма всех чисел в этом наборе, делённая на их количество.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n).$$

## 17.11. Среднеквадратичное значение

**Среднеквадратичное значение** - компонент. Предназначен для получения среднеквадратичного значения элементов массива, получаемая определением среднего значения множества возведенных в квадрат величин.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- {Y} - входной массив.

➤ Выходные

- Число (Yn) - полученное среднеквадратичное значение элементов входного массива.

**Настраиваемые свойства:**

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

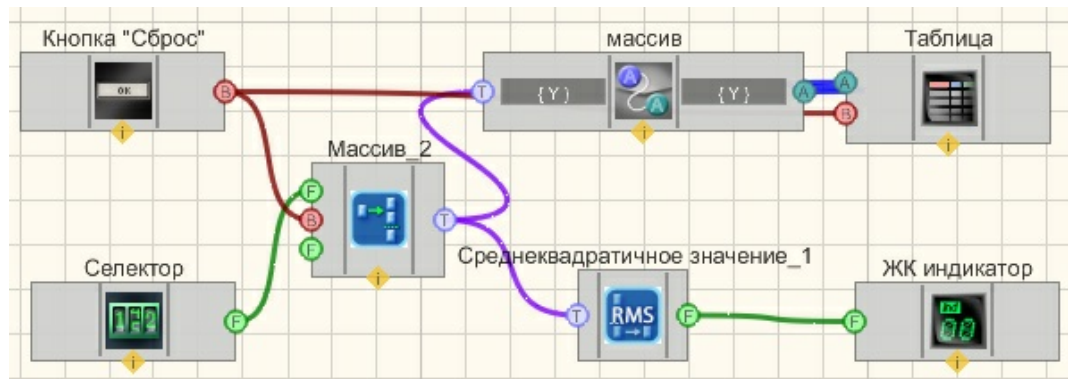
➤ Частные свойства:

Не имеет.



*Пример*

*Проект в SCADA ZETView*



В данном проекте показана работа компонента "[Среднеквадратичное значение](#)<sup>[444]</sup>".

Массив формируется пользователем, с помощью [селектора](#)<sup>[213]</sup>, с которого массив поступает на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> где преобразуется из числового массива в строковый после чего отправляется на [Таблицу данных](#)<sup>[496]</sup> (чтобы пользователь мог видеть, какие значения он вводит в массив). Также с [Формировщика массивов \(число-массив\)](#)<sup>[424]</sup> массив поступает на компонент [Среднеквадратичное значение](#)<sup>[444]</sup>, а результат выводится на [ЖК индикатор](#)<sup>[348]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#)<sup>[422]</sup> и [Таблице данных](#)<sup>[496]</sup> сбрасывает значения массива и очищает таблицу.

### Результат работы проекта

массив
-16.56
-21.56
-9.06
4.69
21.56
40

Ввод массива: **00040.00**

Среднеквадратичное значение: **00023.87**

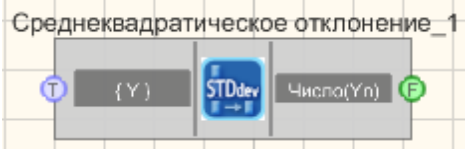
**Сброс**

Посмотреть пример непосредственно в ZETView

## 17.12. Среднеквадратичное отклонение

**Среднеквадратичное отклонение** - компонент. Предназначен для получения значения среднеквадратичного отклонения элементов массива.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - контакт, на который подаются значения массива, исходя из которых будет рассчитываться среднеквадратичное отклонение.

#### ➤ Выходные:

- Число(Yn) - контакт, с которого поступает значение среднеквадратичного отклонения элементов.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства:

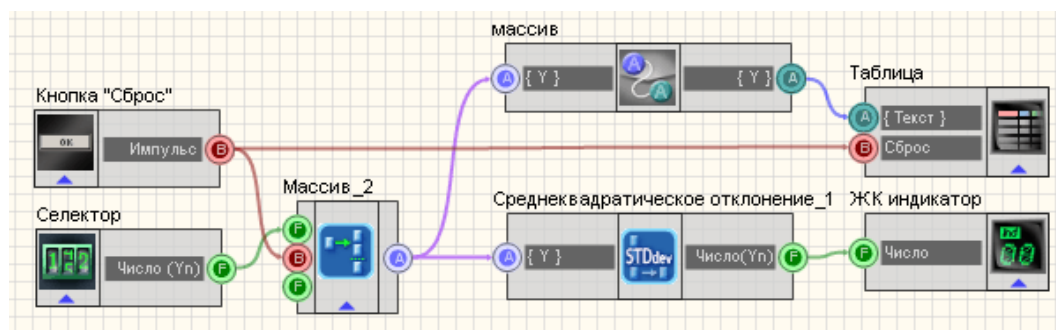
Не имеет.



### *Пример*

### *Проект в SCADA ZETView*

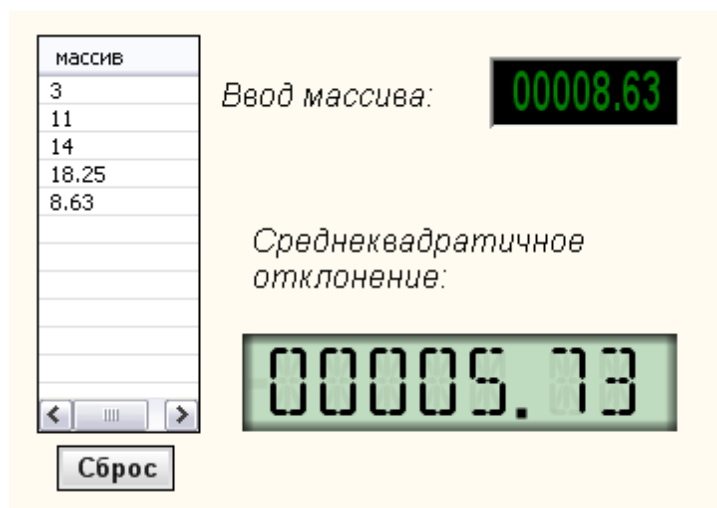




В данном проекте показана работа компонента "[Среднеквадратичное отклонение](#)<sup>[446]</sup>".

Массив формируется пользователем, с помощью [селектора](#)<sup>[213]</sup>, с которого массив поступает на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> где преобразуется из числового массива в строковый после чего отправляется на [Таблицу данных](#)<sup>[496]</sup> (чтобы пользователь мог видеть, какие значения он вводит в массив). Также с [Формировщика массивов \(число-массив\)](#)<sup>[424]</sup> массив поступает на компонент [Среднеквадратичное отклонение](#)<sup>[446]</sup>, а результат выводится на [ЖК индикатор](#)<sup>[348]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#)<sup>[422]</sup> и [Таблице данных](#)<sup>[496]</sup> сбрасывает значения массива и очищает таблицу.

### Результат работы проекта



[Посмотреть пример непосредственно в ZETView](#)



### Математическое описание

Среднеквадратичное отклонение - равно корню квадратному из дисперсии случайной величины. Среднеквадратичное отклонение используют при расчёте стандартной ошибки среднего арифметического, при построении доверительных интервалов, при статистической проверке гипотез, при измерении линейной взаимосвязи между случайными величинами.

Среднеквадратичное отклонение:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2};$$

Компонент "Среднеквадратичное отклонение" рассчитывает значение среднеквадратично отклонения используя стандартное отклонение (несмещённая оценка среднеквадратичного отклонения случайной величины  $x$  относительно её математического ожидания):

$$s = \sqrt{\frac{n}{n-1} \sigma^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2};$$

где  $\sigma^2$  — дисперсия;  $x_i$  —  $i$ -й элемент выборки;  $n$  — объём выборки;  $\bar{x}$  — среднее арифметическое выборки:

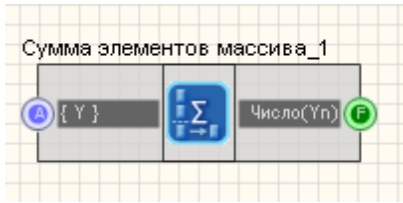
$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n).$$

Измеряется в единицах измерения самой случайной величины.

## 17.13. Сумма элементов массива

Сумма элементов массива - компонент. Вычисляет сумму всех элементов массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- {Y} - Входной массив.

## ➤ Выходные:

- ЧислоY(n) - Сумма элементов массива.

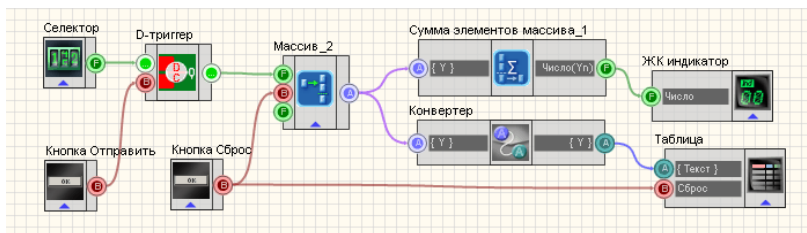
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

Не имеет.

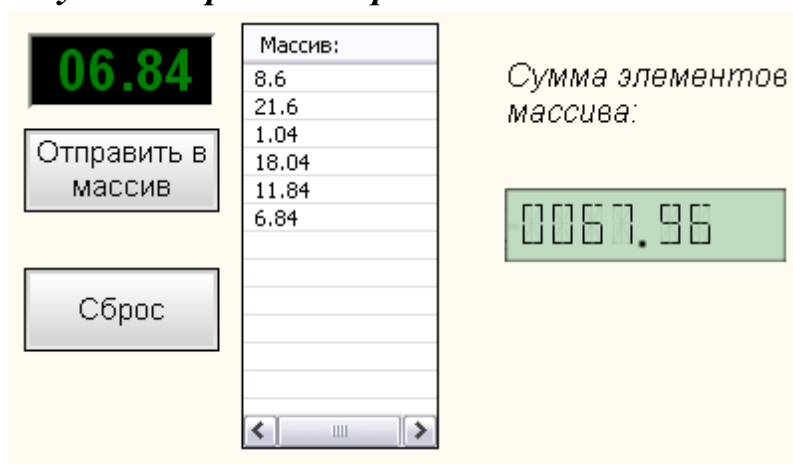
**Пример****Проект в SCADA ZETView**

В данной схеме показана работа компонента Сумма элементов массива. На вход {Y} компонента Сумма элементов массива подается массив, сформированный компонентом [Формировщик массива](#)<sup>424</sup>. Элементы массива формируются с помощью [Селектора](#)<sup>213</sup>, на котором мы можем вручную выставить нужное нам число, затем число с селектора отправляется на [D-триггер](#)<sup>384</sup>. С помощью не фиксирующей кнопки

"Отправить в массив" триггеру дается команда передать значение на [Селекторе](#)<sup>[213]</sup> [Формировщику массива](#)<sup>[424]</sup>. Кнопка сброса, также не фиксирующая кнопка, обнуляет [Формировщик массива](#)<sup>[424]</sup>, на случай если понадобится сформировать массив заново. Далее компонент Сумма элементов массива складывает элементы полученного массива и передает его на выход Число(Уп). После этого число поступает на [Цифровой индикатор](#)<sup>[362]</sup>.

Для большей наглядности работы схемы к ней также подключены [Конвертер \(числ.массив - стр.массив\)](#)<sup>[378]</sup> и [Таблица данных](#)<sup>[496]</sup>, которые позволяют записывать вводимый нами массив в таблицу.

### Результат работы проекта



[Посмотреть пример непосредственно в ZetView](#)

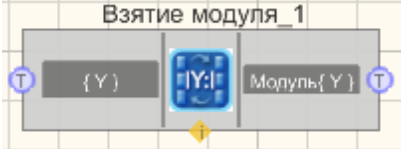
## Глава 18.Массив-Массив

Enter topic text here.

### 18.1.Взятие модуля

**Взятие модуля** - компонент. Предназначен для получения модуля элементов входного массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входы:

- {Y} - контакт, на который нужно подавать входной массив.

#### ➤ Выходные:

- Модуль {Y} - контакт, с которого поступает массив, элементы которого равны модулю элементов входного массива.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

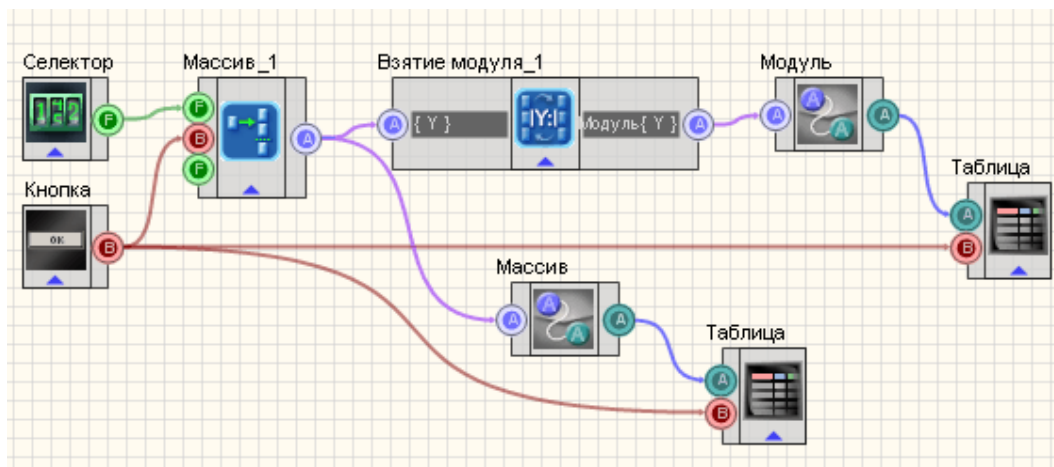
#### ➤ Частные свойства:

Не имеет.



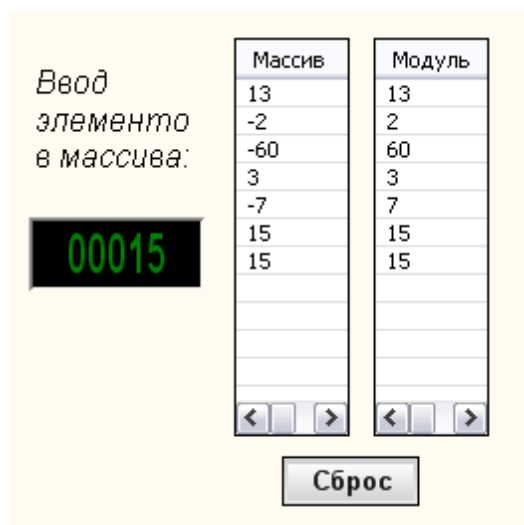
## Пример

### Проект в SCADA ZETView



В данном проекте показана работа компонента ["Взятие модуля"](#)<sup>[450]</sup>. Массив формируется пользователем с помощью [селектора](#)<sup>[213]</sup> и [Формировщика массивов \(число-массив\)](#)<sup>[424]</sup> с которого поступает на компонент ["Взятие модуля"](#)<sup>[450]</sup> и на [Конвертер \(числ.мас.-стр.мас.\)+](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. На выходе с компонента ["Взятие модуля"](#)<sup>[450]</sup> результат также поступает на [Конвертер \(числ.мас.-стр.мас.\)+](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. Поэтому мы можем сравнить при запуске программы исходный массив и результат после работы компонента ["Взятие модуля"](#)<sup>[450]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#)<sup>[422]</sup> и [Таблицам данных](#)<sup>[496]</sup> сбрасывает значения массива и очищает таблицы.

### Результат работы проекта



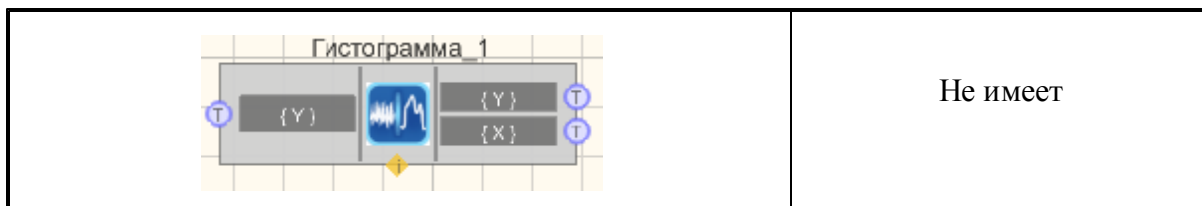
Посмотреть пример непосредственно в ZETView

## 18.2. Гистограмма

**Гистограмма** - компонент. Предназначен для расчета выходного массива значений для построения гистограммы.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------

**Параметры:**

## ➤ Входные:

- {Y} - контакт, на который нужно подавать входной массив.

## ➤ Выходные

- {Y} - выходной массив.
- {X} - размерность массива.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

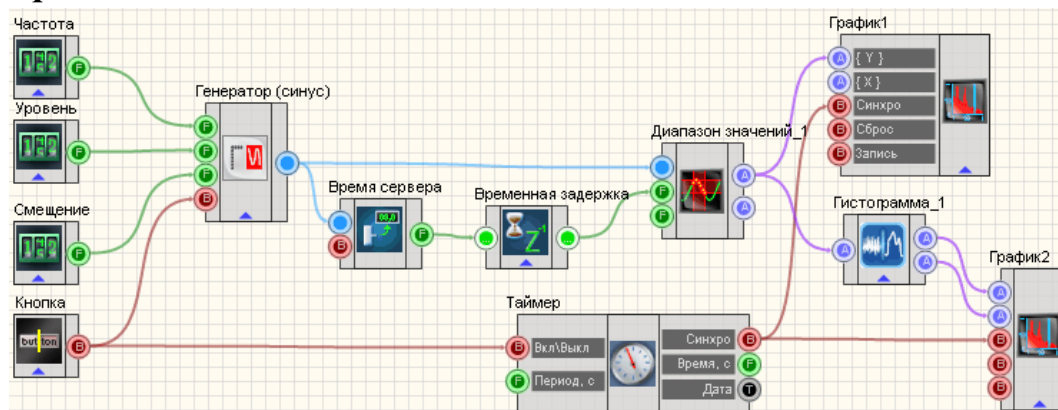
- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Start (0) - устанавливается начальное значение расчета гистограммы.
- Finish (10) - конечное значение расчета гистограммы.
- Quantity (10) - количество полос расчета гистограммы.
- Norm (false) - разрешение нормировки.

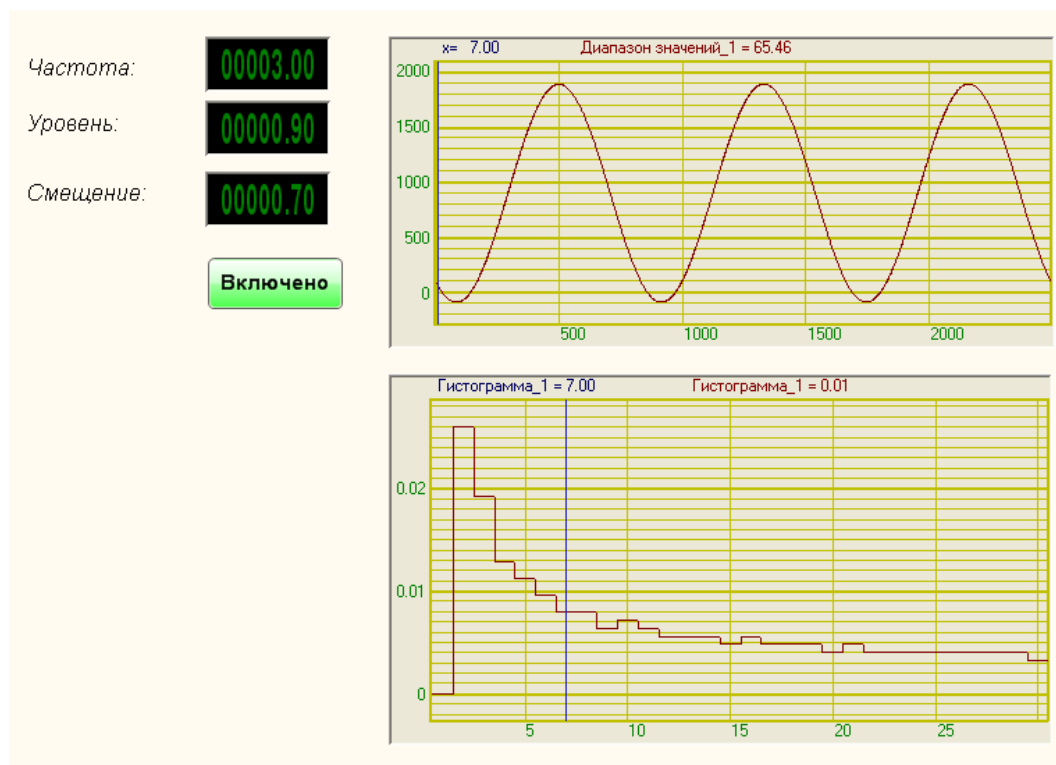
**Пример**

**!** Для работы примера необходимо наличие подключенного физического ZET-устройства.

**Проект в SCADA ZETView**

В примере показана работа компонента ["Гистограмма"](#)<sup>[452]</sup>. Настроив входной сигнал с генератора, данные выводятся на два графика: график сигнала и график гистограммы. Настройка генератора ["Генератор \(синус\)"](#)<sup>[247]</sup> производится с помощью трёх селекторов, которые позволяют настроить частоту, уровень и смещение соответственно. Сигнал с генератора поступает на компонент ["Диапазон значений"](#)<sup>[150]</sup>, который позволяет измерить мгновенные значения сигнала в указанный промежуток времени. Величину промежутка времени компонент ["Диапазон значений"](#)<sup>[150]</sup> получает на входной контакт с компонента ["Время сервера"](#)<sup>[180]</sup>, который специально приспособлен для этой задачи. Можно заметить, что к компоненту ["Время сервера"](#)<sup>[180]</sup> подключен компонент ["Временная задержка"](#)<sup>[178]</sup> - в нем нет необходимости, но желательно использовать его, чтобы не возникло рассинхронизации на компоненте ["Диапазон значений"](#)<sup>[150]</sup> между получаемыми мгновенными значениями сигнала и временем сервера. Далее компонент ["Диапазон значений"](#)<sup>[150]</sup> формирует на выходе два массива: массив значений сигнала и массив индексов. Полученный массив выводится на [график](#)<sup>[351]</sup> - таким образом получается график сигнала. Также данные с ["Диапазон значений"](#)<sup>[150]</sup> поступают на компонент гистограмма, где формируется массив значений гистограммы и массив индексов. С ["Гистограммы"](#)<sup>[452]</sup> массив также попадает на [график2](#) - так получается график гистограммы. Графики синхронизируются с помощью [таймера](#)<sup>[183]</sup>.

### Результат работы проекта





Данный пример непосредственно в ZETView

## 18.3. Дифференцирование массива

<Раздел находится в разработке>

**Дифференцирование массива** - компонент. Компонент дифференцирует массив.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входы:

- {Y} - контакт, на который нужно подавать входной массив.

➤ Выходы:

- A{Y} - контакт, с которого поступает дифференцированный массив.
- 

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства:

Не имеет.

## 18.4. Инверсия массива

**Инверсия массива** - компонент. Инвертирует входной массив.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- {Y} - контакт, на который нужно подавать входной массив.

## ➤ Выходные:

- A{Y} - контакт, с которого поступает инверсный массив.

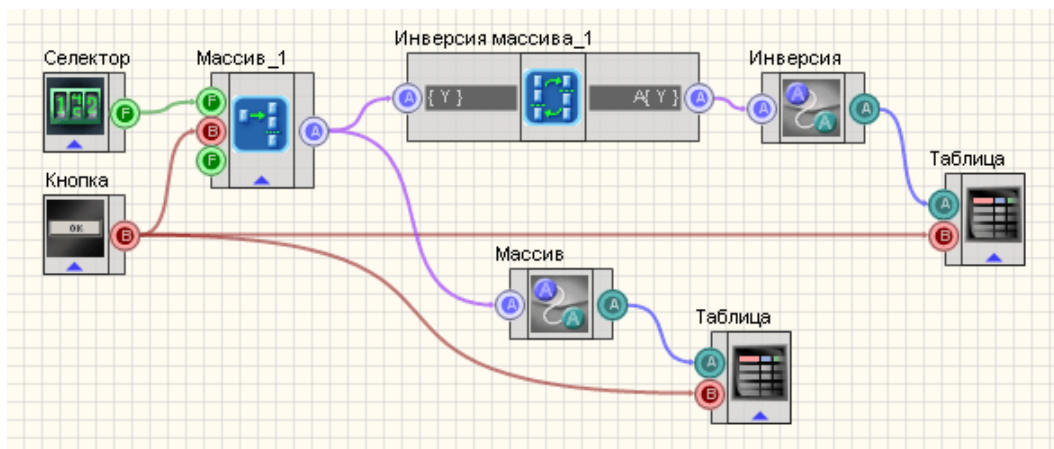
**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства:

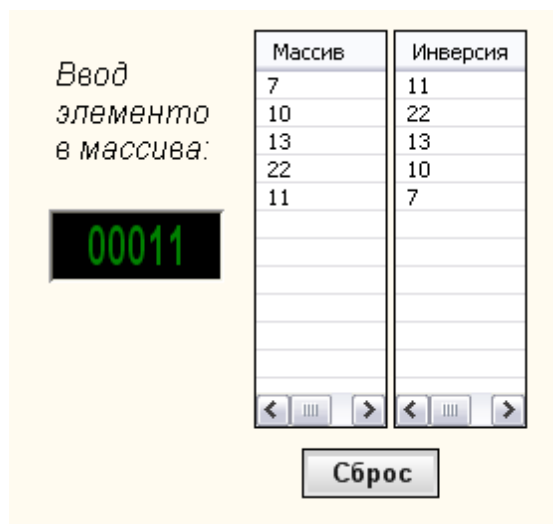
Не имеет.

**Пример****Проект в SCADA ZETView**

В данном проекте показана работа компонента ["Инверсия массива"](#)<sup>[455]</sup>. Массив формируется пользователем с помощью [селектора](#)<sup>[213]</sup> и [Формировщика массивов \(число-массив\)](#)<sup>[424]</sup> с которого поступает на компонент ["Инверсия массива"](#)<sup>[450]</sup> и на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. На выходе с компонента ["Инверсия массива"](#)<sup>[455]</sup> результат также поступает на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. Поэтому мы можем сравнить при запуске программы исходный

массив и результат, после работы компонента "[Инверсия массива](#)".  
[Кнопка без фиксации](#) "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#) и [Таблицам данных](#) сбрасывает значения массива и очищает таблицы.

### Результат работы проекта



Посмотреть пример непосредственно в ZETView

## 18.5. Интегрирование массива

**Интегрирование массива** - компонент. Инвертирует входной массив.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\{Y\}$  - контакт, на который подается входной массив.
- Выходные:
  - $A\{Y\}$  - контакт, с которого поступает интегрированный массив.

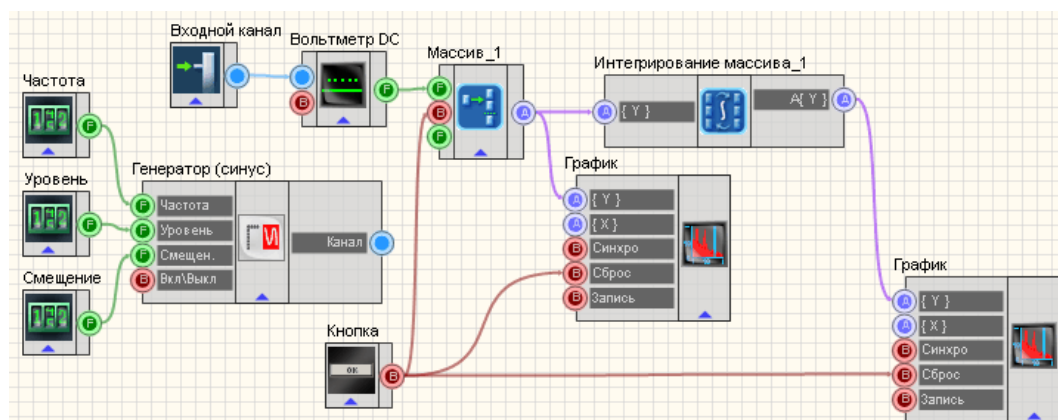
### Настраиваемые свойства:

- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>[92]</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):
  - dTime (1) - интервал между отсчетами, сек.
  - MeanSubstruction (false) - вычитание постоянной составляющей



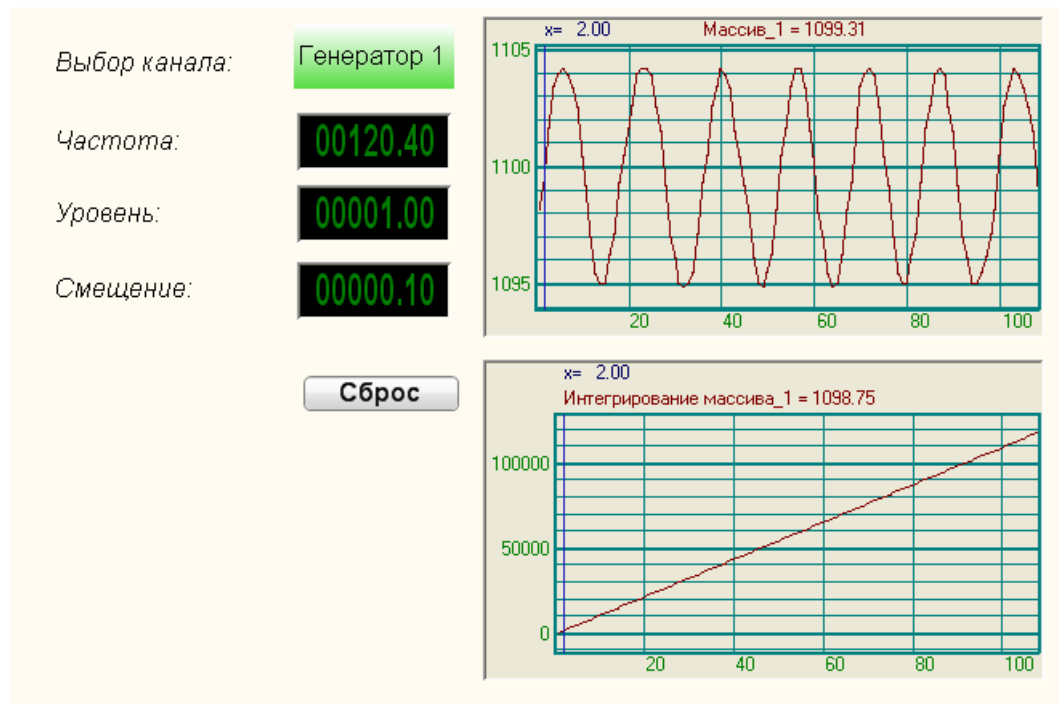
## Пример

### Проект в SCADA ZETView



В данном проекте показана работа компонента ["Интегрирование массива"](#)<sup>[457]</sup>. Массив формируется пользователем с помощью [селектора](#)<sup>[213]</sup> и [Формировщика массивов \(число-массив\)](#)<sup>[424]</sup> с которого поступает на компонент ["Интегрирование массива"](#)<sup>[457]</sup> и на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. На выходе с компонента ["Интегрирование массива"](#)<sup>[457]</sup> результат также поступает на [Конвертер \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup> [Таблица данных](#)<sup>[496]</sup>. Поэтому мы можем сравнить при запуске программы исходный массив и результат, после работы компонента ["Интегрирование массива"](#)<sup>[457]</sup><sup>[450]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> "Сброс", подсоединенная к [Формировщику массивов \(текст-массив\)](#)<sup>[422]</sup> и [Таблицам данных](#)<sup>[496]</sup> сбрасывает значения массива и очищает таблицы.

### Результат работы проекта



Посмотреть пример непосредственно в ZETView

## 18.6. Медианный фильтр

<Раздел находится в разработке>

**Медианный фильтр** - компонент. Предназначен для эффективной обработки сигналов, подверженных воздействию импульсных помех.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входы:

- {Y} - контакт, на который нужно подавать входной массив.

➤ Выходные

- A{Yn} - контакт, с которого поступает фильтрованный медианным фильтром

массив.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):


- WindowLen (3) - ширина окна медианного фильтра

## 18.7.Метод наименьших квадратов

<Раздел находится в разработке>

**Метод наименьших квадратов** - компонент. Предназначен для оценки неизвестных величин по результатам измерений, содержащих случайные ошибки.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

➤ Входные:

- {Y} - входной массив Y.
- {X} - входной массив X.

➤ Выходные

- {Y} - выходной массив.
- {X} - размерность массива.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- type (Линейная) - зависимость:
  - Линейная.


- Степенная.
- Показательная.
- Дробно-линейная.
- Дробно-рациональная.
- Логарифмическая.
- Гиперболическая.

## 18.8. Нормализация массива

<Раздел находится в разработке>

**Нормализация массива** - компонент. Предназначен для нормализации массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\{Y\}$  - контакт, на который нужно подавать входной массив.

➤ Выходные:

- $A\{Y\}$  - контакт, с которого поступает нормализованный массив.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

➤ Частные свойства:

Не имеет.

## 18.9. Обратное преобразование Фурье

<Раздел находится в разработке>

**Обратное преобразование Фурье** - компонент.

Производит быстрое обратное преобразование Фурье, где на входе массивы действительных и мнимых составляющих, а на выходе массив.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\text{Re}(F\{Y\})$  - действительные коэффициенты преобразования Фурье.
- $\text{Im}(F\{Y\})$  - мнимые коэффициенты преобразования Фурье.

➤ Выходные

- $\{Y\}$  - выходной массив.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства:

Не имеет.

## 18.10. Огибающая

<Раздел находится в разработке>

**Огибающая** - компонент. Кривая называется огибающей семейства кривых, зависящих от параметра, если она в каждой своей точке касается хотя бы одной кривой семейства и каждым своим отрезком касается бесконечного множества этих кривых.



**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- {Y} - входной массив.

## ➤ Выходные

- {Y} - выходной массив.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

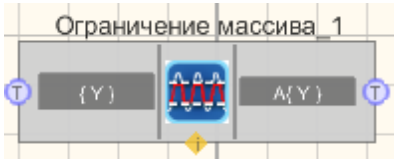
type - зависимость:

1. линейная.
2. степенная.
3. показательная.
4. дробно-линейная.
5. дробно-рациональная.
6. логарифмическая.
7. гиперболическая.

## 18.11. Ограничение массива

**Ограничение массива** - компонент. Ограничивает массив по выставленному уровню и ширине уровня, которая определяет границы. Элементы массива не попадающие в указанный диапазон заменяются либо значением уровня, либо ближайшим граничным значением.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - контакт, на который нужно подавать входной массив.

#### ➤ Выходные:

- A{Y} - контакт, с которого поступает ограниченный массив.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

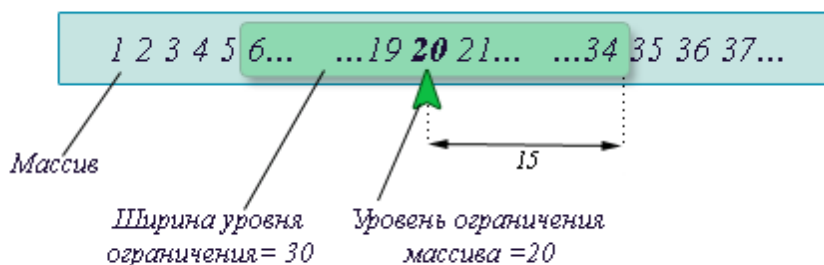
- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

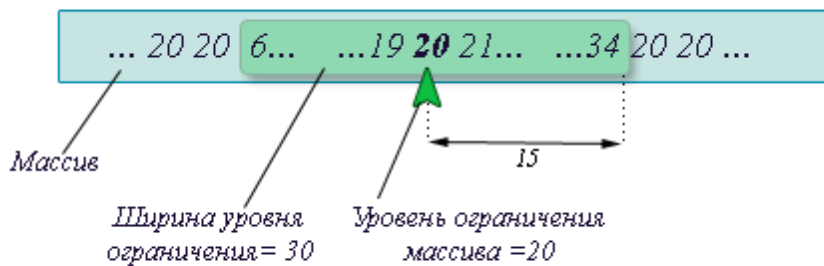
- Level (0) - амплитуда выходного сигнала (в единицах измерения канала).
- LevelWidth (1) - ширина уровня (в единицах измерения канала).
- changetype (экстремум) - вид замены.
  1. экстремум.
  2. уровень.

### Особенности работы компонента:

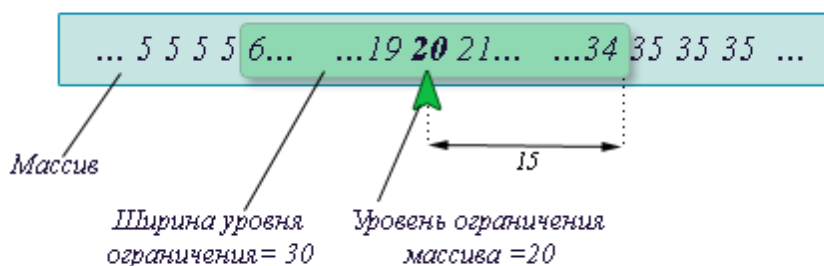
Рассмотрим входной массив:



После прохождения массива через компонент "Ограничение массива" с настройкой замены по уровню получим массив:



Как видно из рисунка, все значения, не попадающие в указанный диапазон, включая граничные, заменяются значением уровня ограничения, указанного пользователем. Если поставить настройку замены "экстремум", то значения, не попавшие в указанный диапазон, будут заменены на ближайший экстремум:



В данном примере это 5 и 35.

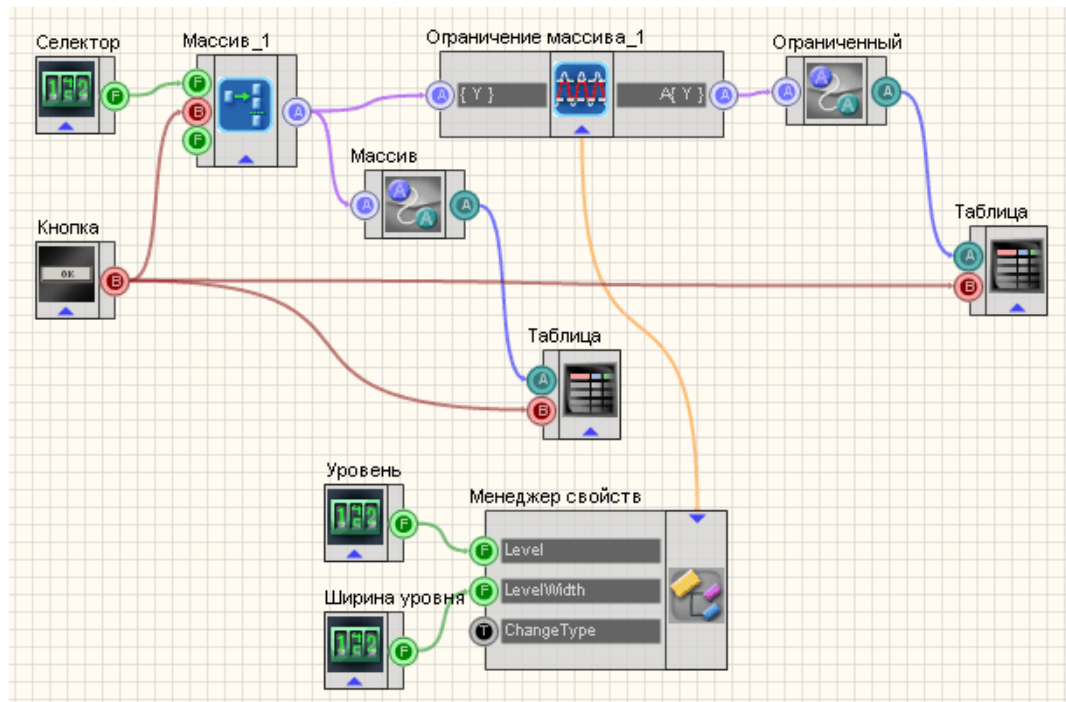


## **Пример**

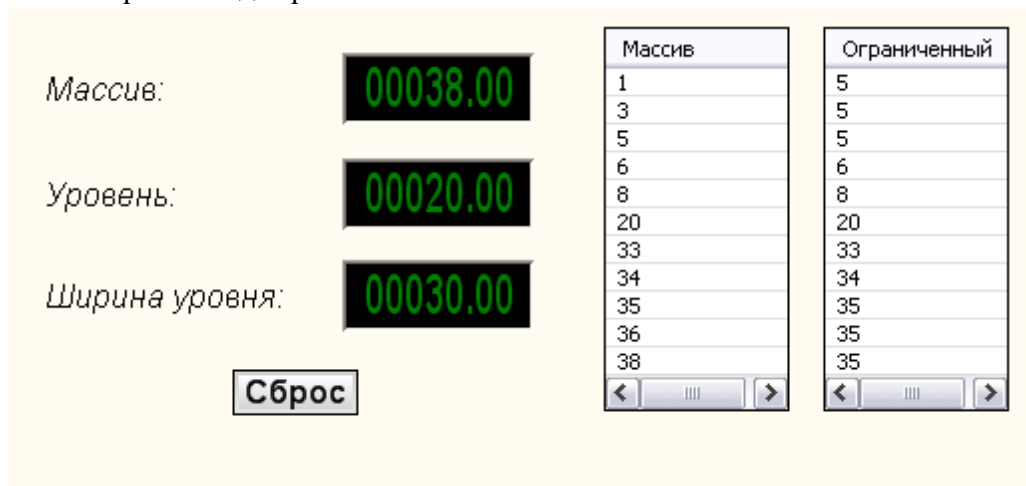
Пример 1

Настройка замены: "Экстремум"

**Проект в SCADA ZETView**



В данной схеме показана работа компонента [Ограничение массива](#)<sup>[463]</sup>. С помощью [Селектора](#)<sup>[213]</sup> вводятся элементы массива, которые затем поступают на [формировщик массивов](#)<sup>[424]</sup>. Далее сформированный массив поступает на вход компонента [Ограничение массива](#)<sup>[463]</sup> а также конвертируется в строковый массив, с помощью [Конвертера \(числ.мас.-стр.мас.\)](#)<sup>[378]</sup>, и отображается в [Таблице данных](#)<sup>[496]</sup>. Выходной ограниченный массив с компонента [Ограничение массива](#)<sup>[463]</sup> также конвертируется и выводится в [Таблицу данных](#)<sup>[496]</sup>. Также два селектора "Уровень" и "Ширина уровня" подключены к "Ограничение массива"<sup>[463]</sup> через [Менеджер свойств](#)<sup>[534]</sup>, позволяют пользователю настраивать соответствующие характеристики. Так как изначально у компонента "Ограничение массива"<sup>[463]</sup> нет данных контактов, а только лишь настраиваемые свойства, селекторы подключаются к ним через менеджер свойств.

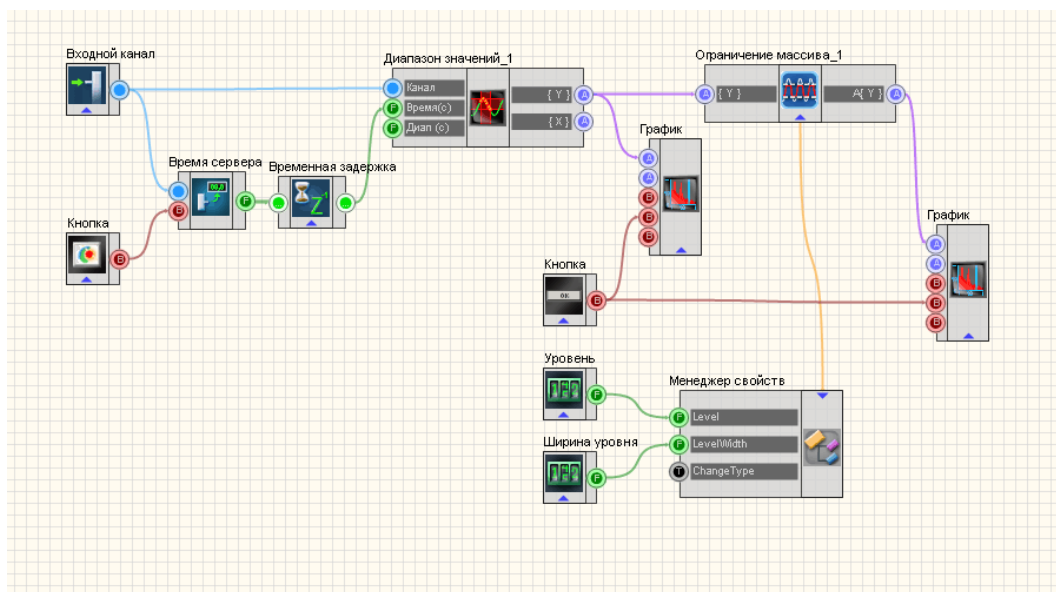


Посмотреть пример непосредственно в ZETView

## Пример 2

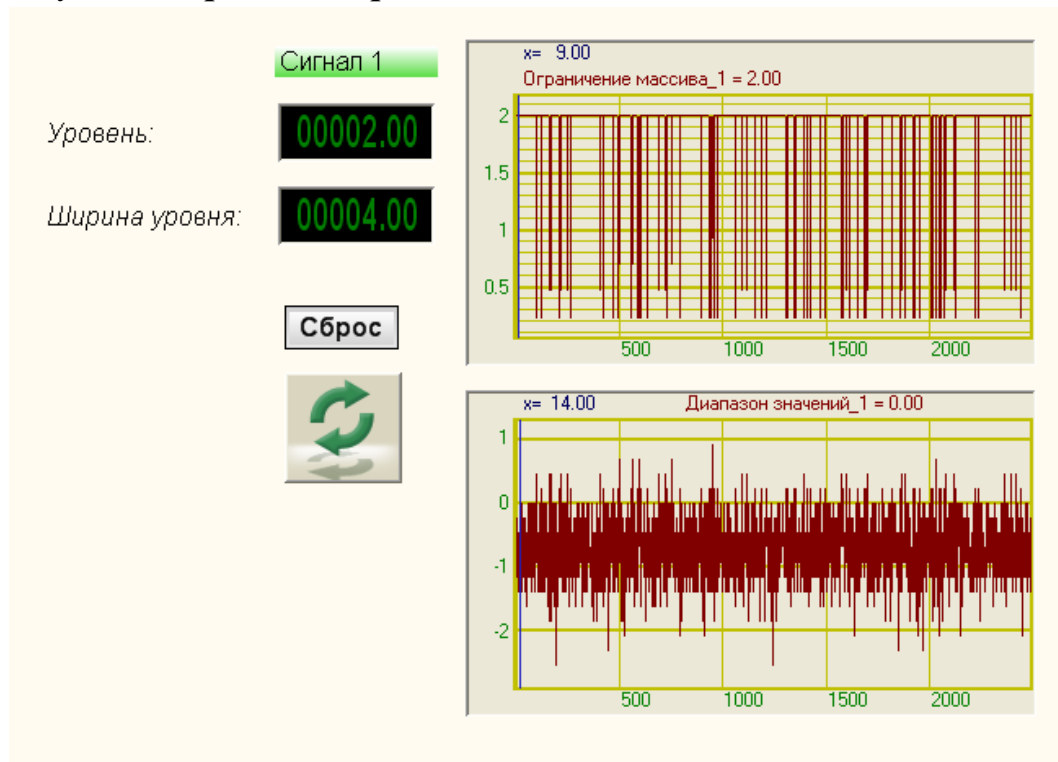
Настройка замены: "Уровень"

### Проект в SCADA ZETView



В данной схеме показана работа компонента [Ограничение массива](#)<sup>[463]</sup>. С помощью компонента "[Входной канал](#)<sup>[227]</sup>" принимается сигнал с физического ZET-устройства, подключенного к компьютеру. Компонент [Диапазон значений](#)<sup>[290]</sup> формирует массив мгновенных значений сигнала в определенный промежуток времени. Время канала на входе компонента [Диапазон значений](#)<sup>[290]</sup> формируется с помощью связки [Время сервера](#)<sup>[180]</sup> + [Временная задержка](#)<sup>[178]</sup>. Притом второй компонент необязателен, благодаря ему значение на входе "Канал" компонента [Диапазон значений](#)<sup>[290]</sup> успевает накапливаться. Подобный ход ограждает проектировщика от ошибки, когда значения времени приходят раньше значений канала. После массив поступает на компонент "[Ограничение массива](#)<sup>[463]</sup>", а также выводится на [График](#)<sup>[490]</sup>. После прохождения компонента "[Ограничение массива](#)<sup>[463]</sup>" также идет вывод на [График](#)<sup>[490]</sup>. Два селектора "Уровень" и "Ширина уровня" подключены к "[Ограничение массива](#)<sup>[463]</sup>" через [Менеджер свойств](#)<sup>[534]</sup>, позволяют пользователю настраивать соответствующие характеристики. Так как изначально у компонента "[Ограничение массива](#)<sup>[463]</sup>" нет данных контактов, а только лишь настраиваемые свойства, селекторы подключаются к ним через менеджер свойств.

## Результат работы проекта



[Посмотреть пример непосредственно в ZETView](#)

## 18.12.Операция с константой

**Операция с константой** - компонент. Выполняет выбранное действие над полученной константой и входным массивом.

При выборе действия в свойствах компонента "Операция с константой", например "сложение", будет совершаться поэлементно с каждым элементом массива.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- $\{Y\}$  - входной массив.
- Константа - значение константы.

## ➤ Выходные

- $A\{Y\}$  - выходной массив

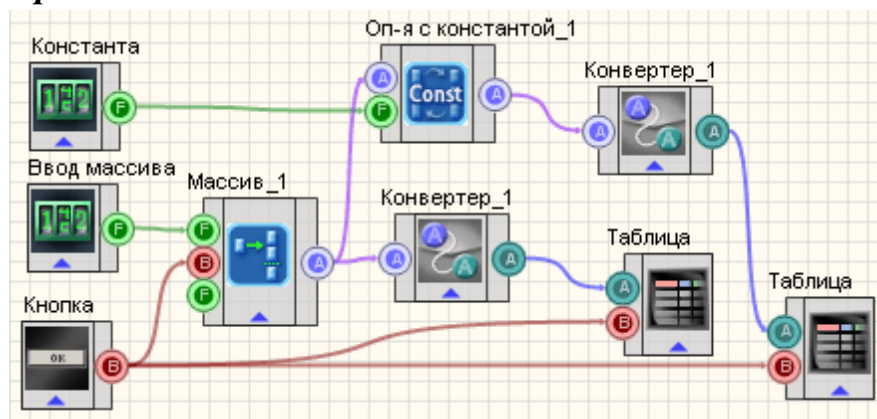
**Настраиваемые свойства:**

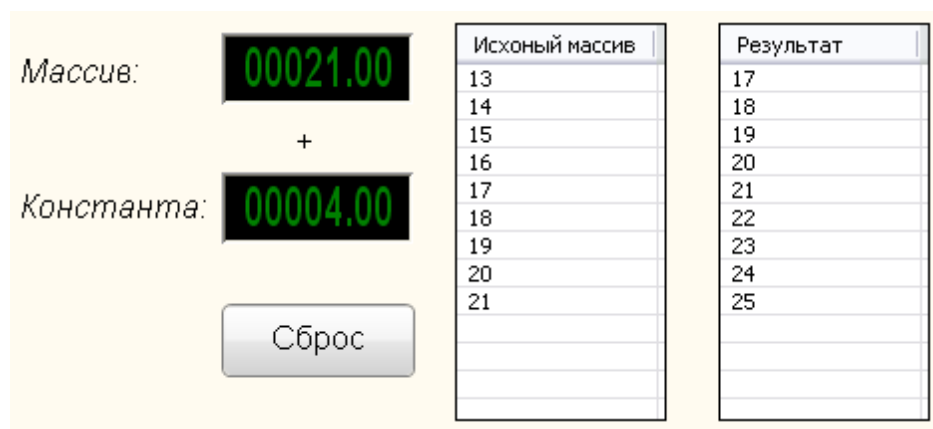
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Constant (0) - константа для арифметических действий над массивом
- operationtype (Сложение) - вариант арифметического действия над массивом:
  1. Сложение.
  2. Вычитание.
  3. Умножение.
  4. Деление.
  5. Константа в степени элементов массива.
  6. Элементы массива в степени константы.
  7. Логарифм элементов по основанию константы.

**Пример****Проект в SCADA ZETView****Результат работы проекта**



Посмотреть пример непосредственно в ZetView

## 18.13.Преобразование координат в декартовы

<Раздел находится в разработке>

**Преобразование координат в декартовы** - компонент. Предназначен для преобразования из полярных координат в декартовы

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\text{Ampl}\{Y\}$  - массив модулей массива комплексных чисел.
- $\text{Phase}\{Y\}$  - массив фаз массива комплексных чисел.

➤ Выходные

- $\text{Re}(F\{Y\})$  - действительные части массива комплексных чисел.
- $\text{Im}(F\{Y\})$  - мнимые части массива комплексных чисел.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

➤ Частные свойства:

Не имеет.

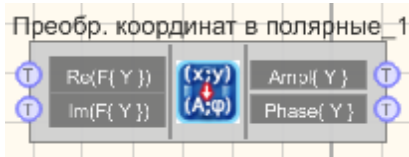


## 18.14. Преобразование координат в полярные

<Раздел находится в разработке>

**Преобразование координат в полярные** - компонент. Предназначен для преобразования из декартовых координат в полярные.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\text{Re}(F\{Y\})$  - действительные части массива комплексных чисел.
- $\text{Im}(F\{Y\})$  - мнимые части массива комплексных чисел.

➤ Выходные

- $\text{Ampl}\{Y\}$  - выходной массив модулей массива комплексных чисел.
- $\text{Phase}\{Y\}$  - выходной массив фаз массива комплексных чисел.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства:

Не имеет.

## 18.15.Преобразование Фурье

**Преобразование Фурье** - компонент. Производит быстрое преобразование Фурье над входным массивом, выдавая на выходе массивы действительных и мнимых составляющих.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- $\{Y\}$  - входной массив.

#### ➤ Выходные:

- $Re(F\{Y\})$  - действительные коэффициенты преобразования Фурье.
- $Im(F\{Y\})$  - мнимые коэффициенты преобразования Фурье.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

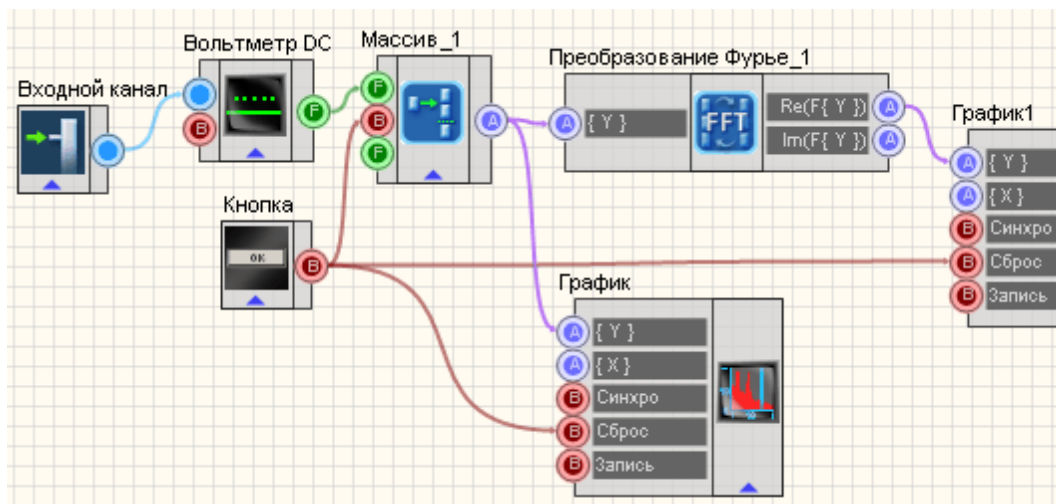
#### ➤ Частные свойства:

Не имеет.



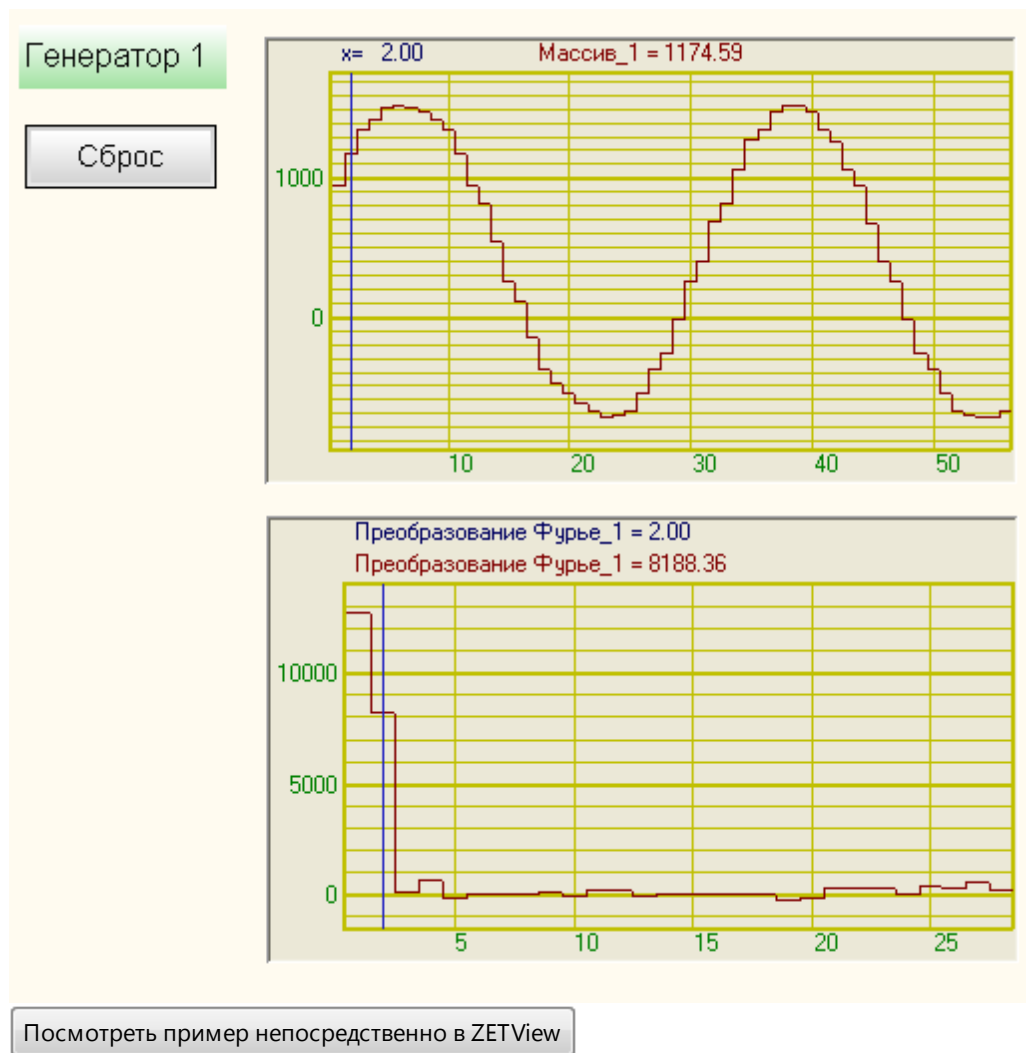
### *Пример*

### *Проект в SCADA ZETView*



В данном проекте, используется виртуальный прибор ZETLab "Генератор сигн" которого принимается сигнал посредством компонента ["Измерительный к"](#) 227  
 Значения с генератора поступают на ["Вольтметр DC"](#)<sup>288</sup>, после чего изм  
 значения напряжений поступают на компонент "Формировщик массивов  
 массив)", далее массив чисел поступает на ["График"](#)<sup>490</sup>, тем самым исходный  
 выводится на первый график, и на компонент ["Преобразование Фурье"](#) 472  
 подвергается БПФ и также выводится на второй график.

### *Результат работы проекта*



## 18.16. Ударный спектр

**Ударный спектр** - компонент. Ударный спектр (спектр ударного отклика, shock response spectrum) – зависимость максимальных откликов на возмущающее воздействие системы из резонаторов с одной степенью свободы, выстроенных в порядке возрастания собственных частот.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- {Y} - входной массив с данными.

#### ➤ Выходные:

- SRS - спектр ударного отклика.
- Freqs - частотный ряд.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- SamplingRate (25000) – частота дискретизации входных данных;
- FirstFrequency (10) – первая частота в частотном ряду;
- LastFrequency (5000) – последняя частота в частотном ряду;
- DampingKoeff (0,05) – коэффициент затухания колебательных систем;
- FreqsInOctave (16) – количество частотных полос в одной октаве.

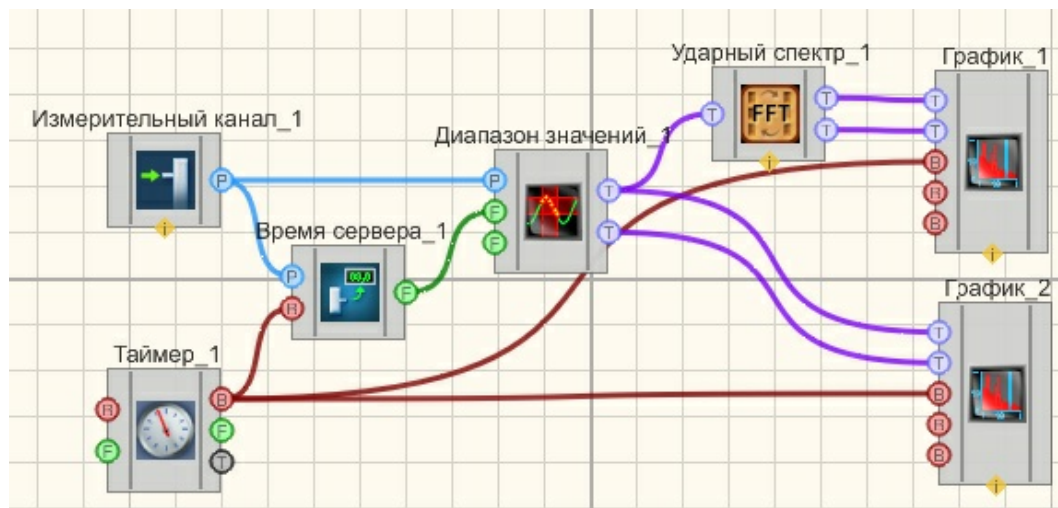
### • Примечания:

- Последняя частота в частотном ряду не должна быть больше  $j$  от частоты дискретизации.
- Коэффициент затухания по умолчанию равен 0,05.
- Алгоритм очень ресурсоёмкий, не стоит подавать на вход слишком часто большие объёмы данных.

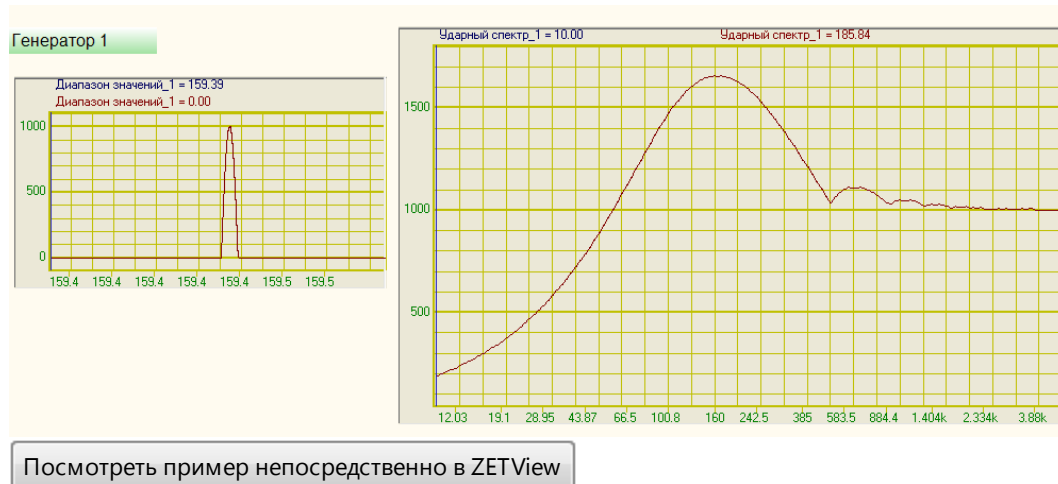


### *Пример*

### *Проект в SCADA ZETView*



### Результат работы проекта



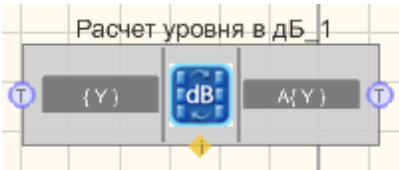
## 18.17. Расчет уровня в дБ

<Раздел находится в разработке>

Расчет уровня в дБ - компонент.

Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------

	Не имеет
---	----------

**Параметры:**

## ➤ Входящие:

- {Y} - контакт, на который нужно подавать входной массив.

## ➤ Выходные:

- A{Y} - контакт, с которого поступает расчет значения в дБ.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Reference (0.001) - значение для расчета уровня в дБ.
- factor (20) - множитель для расчета в дБ.

## 18.18. Сокращение массива

**Сокращение массива** - компонент. Сокращает массив, путем отсечения N-го количества элементов справа или слева, где N - размер сокращения массива.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- {Y} - контакт, на который подается входной массив.

## ➤ Выходные:

- $A\{Y\}$  - контакт, с которого поступает сокращенный массив.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

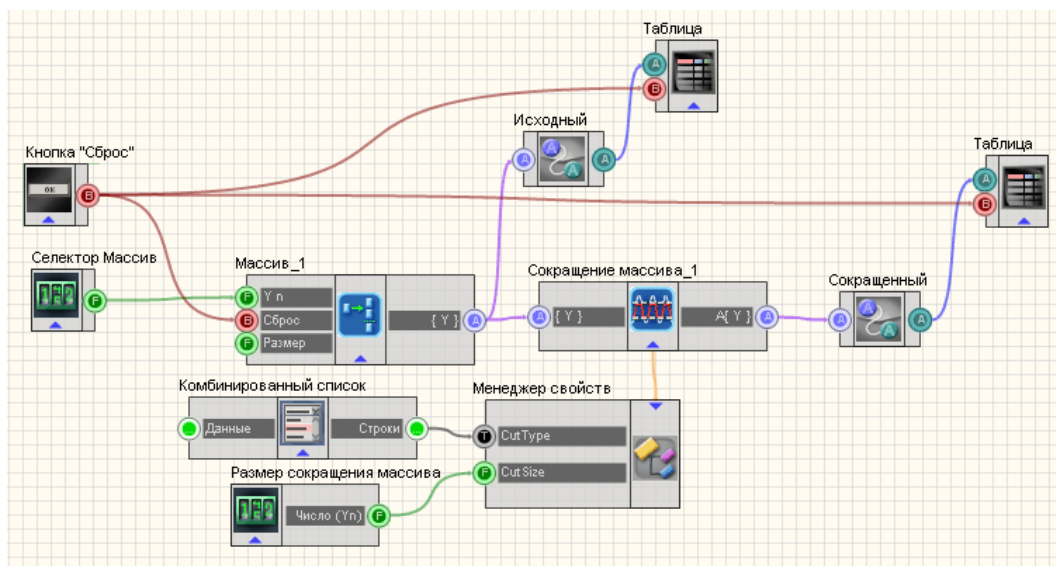
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- cuttype (Справа) - Вариант сокращения массива. На выбор принимает два значения: Слева и Справа. Посредством данной настройки выбирается с какой стороны будут отсечены элементы массива.
- CutSize (0) - Размер сокращения массива. Принимает любое целочисленное значение. Означает количество отсекаемых элементов массива.



## Пример

### Проект в SCADA ZETView



В данной схеме показана работа компонента [Сокращение массива](#)<sup>[477]</sup>.

С помощью [Селектора](#)<sup>[213]</sup> вводятся элементы массива, которые затем поступают на [формировщик массивов](#)<sup>[424]</sup>. Далее сформированный массив поступает на вход компонента [Сокращение массива](#)<sup>[477]</sup>, также конвертируется в строковый массив, с помощью [Конвертера \(числ.мас.-стр. мас.\)](#)<sup>[378]</sup>, и отображается в [Таблице данных](#)<sup>[496]</sup>. Выходной сокращенный массив с компонента [Сокращение массива](#)<sup>[477]</sup> также конвертируется и выводится в [Таблицу данных](#)<sup>[496]</sup>.

[Селектор](#)<sup>[213]</sup> "Размер сокращения массива" и [Комбинированный список](#)<sup>[203]</sup>



"Ширина уровня" подключены к "Сокращение массива<sup>[477]</sup>" через [Менеджер свойств<sup>\[534\]</sup>](#), позволяют пользователю настраивать соответствующие характеристики. Так как изначально у компонента "Сокращение массива<sup>[477]</sup>" нет данных контактов, а только лишь настраиваемые свойства, селекторы подключаются к ним через менеджер свойств.

### Результат работы проекта

**МАССИВ**

Ввод массива: 00010.00

Вариант сокращения массива:

Слева

Размер сокращения массива:

003

Исходный	Сокращенный.
1	4
2	5
3	6
4	7
5	8
6	9
7	10
8	
9	
10	

[Посмотреть пример непосредственно в ZETView](#)

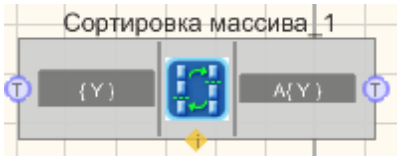
## 18.19.Сортировка массива

<Раздел находится в разработке>

**Сортировка массива** - компонент. Сортирует входной массив, располагая элементы по возрастанию или по убыванию.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------

	Не имеет
---	----------

**Параметры:**

## ➤ Входные:

- {Y} - входной контакт, на который поступает числовой массив.

## ➤ Выходные

- A{Y} - выходной контакт, с которого поступает сортированный выходной массив.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- sorttype (по убыванию) - тип сортировки:
  - по убыванию
  - по возрастанию

**18.20. Фильтрация массива**

<Раздел находится в разработке>

**Фильтрация массива** - компонент.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- $\{Y\}$  - контакт, на который нужно подавать входной массив для фильтрации.

➤ Выходные

- $F\{Y\}$  - контакт, с которого поступает фильтрованный массив.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

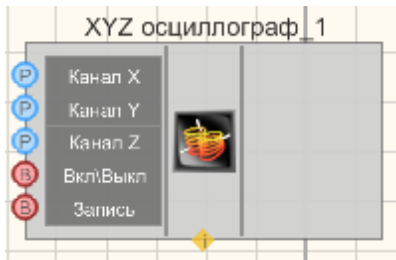
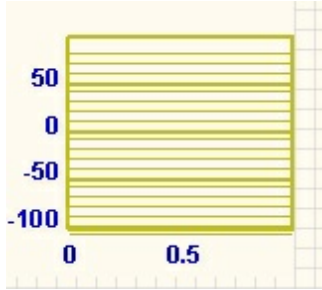
- $f_{Max}$  (100) - максимальная частота среза ФНЧ, Гц
- $f_{Min}$  (10) - минимальная частота среза ФВЧ, Гц
- frequency (25000) - частота дискретизации исходного сигнала, Гц
- filtertype (линейный) - тип фильтра:
  1. линейный.
  2. интегрирующий.
  3. интегрирующий 2.

## Глава 19. Отображение

### 19.1. XYZ осциллограф

**XYZ осциллограф** - компонент. Предназначен для просмотра формы сигнала, измерения мгновенных значений сигнала и отображения параметрической зависимости сигналов.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

**Параметры:**

➤ Входные:

- Канал X - канал данных оси X.
- Канал Y - канал данных оси Y.
- Канал Z - канал данных оси Z.
- Вкл\Выкл - параметр, отвечающий за включение и выключение XYZ-осциллографа.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

➤ Выходные

Не имеет.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

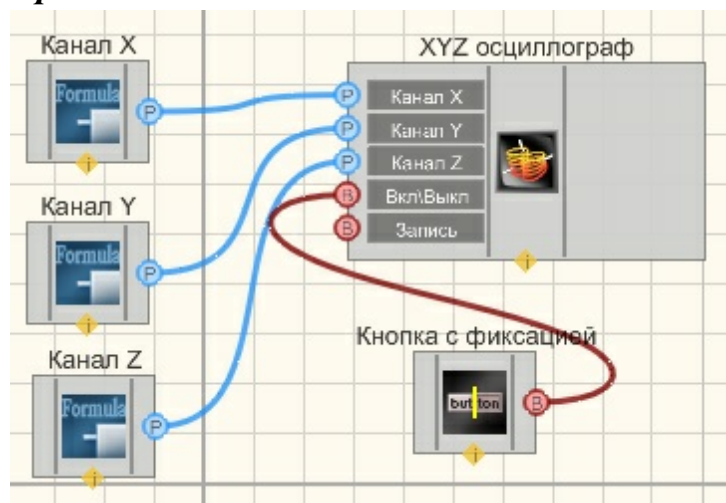
- GridColor (bfbf20) - установка цвета координатной сетки.
- CursorColor (000080) - установка цвета курсора.
- BackColor (ffffb0) - установка цвета заднего фона.
- StartGraphColor (f0f0f0) - установка начального цвета графика.
- EndGraphColor (601010) - установка конечного цвета графика.
- DigitsColor (0000be) - установка цвета надписей на осях.
- LegendColor (009400) - установка цвета шапки
- coordinatesystem ( XT) - система координат:
  1. XT.
  2. YT.
  3. ZT.
  4. XY.
  5. XZ.
  6. YZ.
  7. XYT.
  8. XZT
  9. YZT.
  10. XYZ.
- coordinategrid (Сетка по X и Y) - установка координатной сетки:
  1. Нет сетки.
  2. Сетка по X.
  3. Сетка по Y.
  4. Сетка по X и Y.
- update (1 с) - обновление графика (0.1 и 1 с.)
- interval (1)- интервал отображения, с
- TStart (0) - смещение нуля по оси времени, с
- TInterval (1) - длина отображения по оси времени, с

- Xstart (-100) - начало отображения по оси X, ед. изм.
- Xend (100) - конец отображения по оси X, ед. изм.
- Ystart (-100) - начало отображения по оси Y, ед. изм.
- Yend (100) - конец отображения по оси Y, ед. изм.
- Zstart (-100) - начало отображения по оси Z, ед. изм.
- Zend (100) - конец отображения по оси Z, ед. изм.
- FileName- имя файла для записи результатов
- Activate (true) - состояние работы (включено (true)/выключено (false))

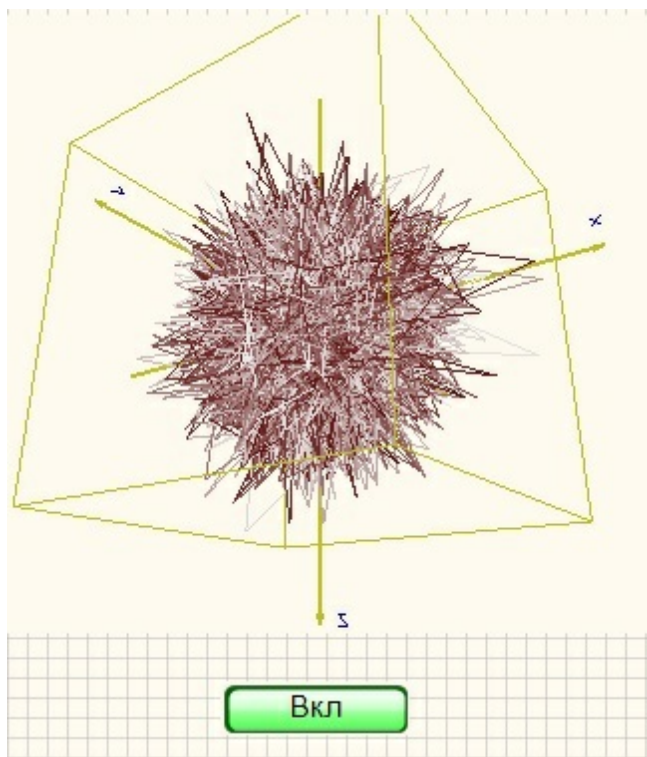


### Пример

#### Проект в SCADA ZETView



#### Результат работы проекта



Данный пример непосредственно в ZETView

## 19.2.XYZ плоттер

<Раздел находится в разработке>

**XYZ плоттер** - компонент. Предназначен для визуализации (оценки формы) взаимных характеристик двух измеряемых величин.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

**Параметры:**

➤ Входы:

- Канал X - канал данных по оси X.
  - Канал Y - канал данных по оси Y.
  - Канал Z - канал данных по оси Z.
  - Очистка - контакт, при подаче на который импульса, производится очистка отображения.
  - Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).
- Выходные  
Не имеет.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- GridColor (bfbf20) - установка цвета координатной сетки.
- CursorColor (000080) - установка цвета курсора.
- BackColor (ffbf0) - установка цвета заднего фона.
- StartGraphColor (f0f0f0) - установка начального цвета графика.
- EndGraphColor (601010) - установка конечного цвета графика.
- DigitsColor (0000be) - установка цвета надписей на осях.
- LegendColor (009400) - установка цвета шапки.
- coordinatesystem (XT) - система координат:
  1. XT.
  2. YT.
  3. ZT.
  4. XY.
  5. XZ.
  6. YZ.
  7. XYT.
  8. XZT.
  9. YZT.
  10. XYZ.
- coordinatagrid (Сетка по X и Y) - установка координатной сетки:
  1. Нет сетки.
  2. Сетка по X.
  3. Сетка по Y.
  4. Сетка по X и Y.
- Autoscale (false) - автомасштаб.
- Xstart (-100) - начало отображения по оси X, ед. изм.
- Xend (100) - конец отображения по оси X, ед. изм.
- Ystart (-100) - начало отображения по оси Y, ед. изм.
- Yend (100) - конец отображения по оси Y, ед. изм.

- Zstart (-100) - начало отображения по оси Z, ед. изм.
- Zend (100) - конец отображения по оси Z, ед. изм.
- FileName - имя файла для записи результатов

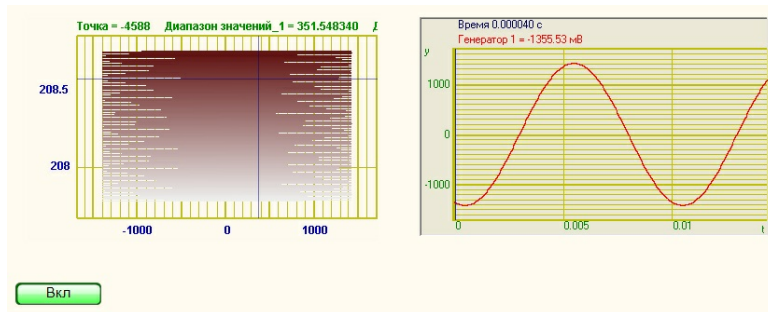


## Пример

### Проект в SCADA ZETView



### Результат работы проекта



Данный пример непосредственно в ZETView

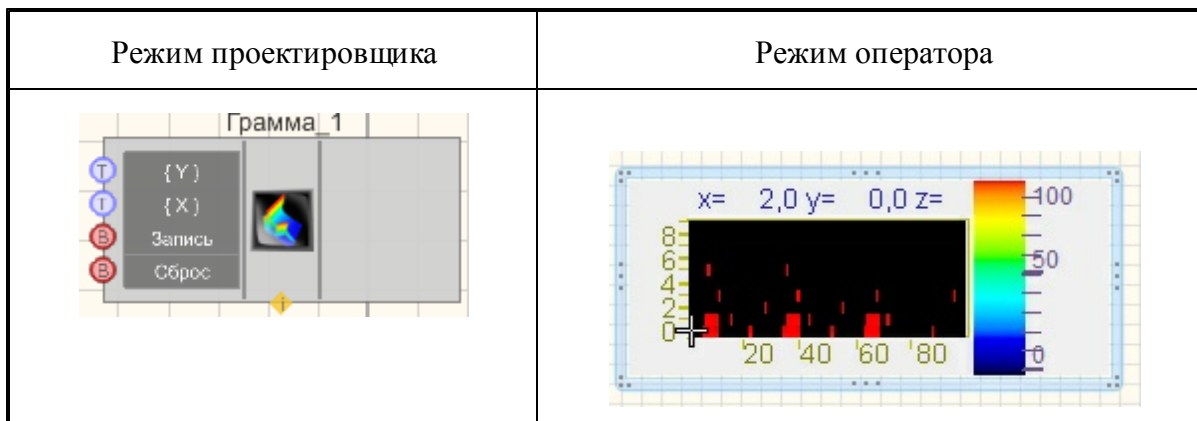
## 19.3.Грамма

<Раздел находится в разработке>

**Грамма**- компонент. Предназначен для отображения численных значений на специально оформленной области в 3х мерном изображении

**Внешний вид компонента:**





### Параметры:

#### ➤ Входные:

- {Y} - контакт, к которому подключается массив со значениями, которые будут отображены по оси ординат.
- {X} - контакт, к которому подключается массив со значениями, которые будут отображены по оси абсцисс.
- Сброс - контакт, при подаче на который импульса, происходит очистка графика.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

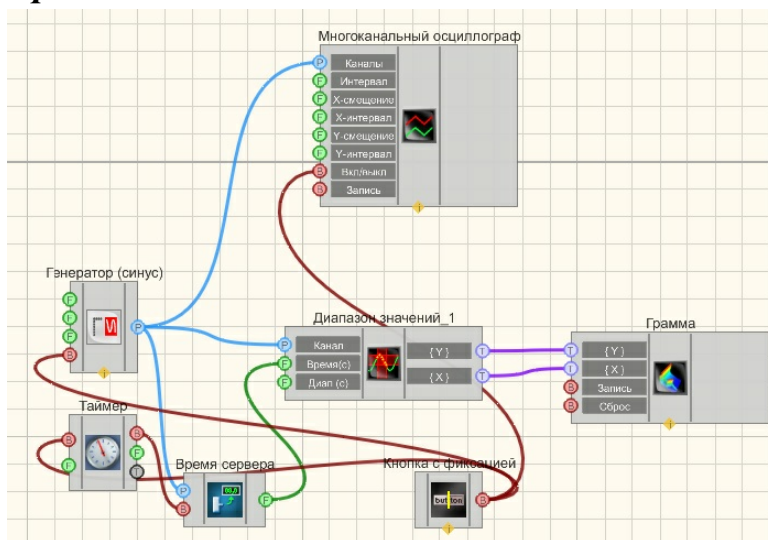
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- allowPicture (false) - разрешение загрузки рисунка из файла.
- PictureFile - установка имени и расположения файла, содержащего рисунок
- LineQuantity (10) - количество отсчётов.
- DeltaT (1) - установка интервала между строками по оси Y (в единицах измерения)
- XAutoscale (true) - включение/выключение автомасштабирования по оси X.
- YAutoscale (true) - включение/выключение автомасштабирования по оси Y.
- XString (Гц) - установка единицы измерения по оси X.
- YString (с) - установка единицы измерения по оси Y.
- ZString (мВ) - установка единицы измерения по оси Z.
- BlackMode (false) - установка отображения в чёрно-белом режиме.
- XAxis (false) - установка разметки по оси X.
- FileName - установка имени и расположения файла, в котором данные будут сохранены в виде отдельного файла.

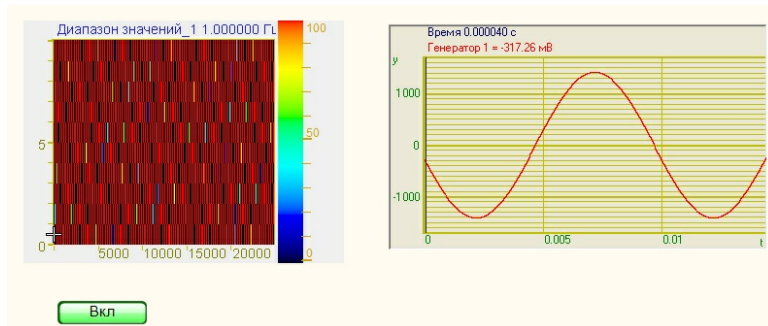


## Пример

### Проект в SCADA ZETView



### Результат работы проекта

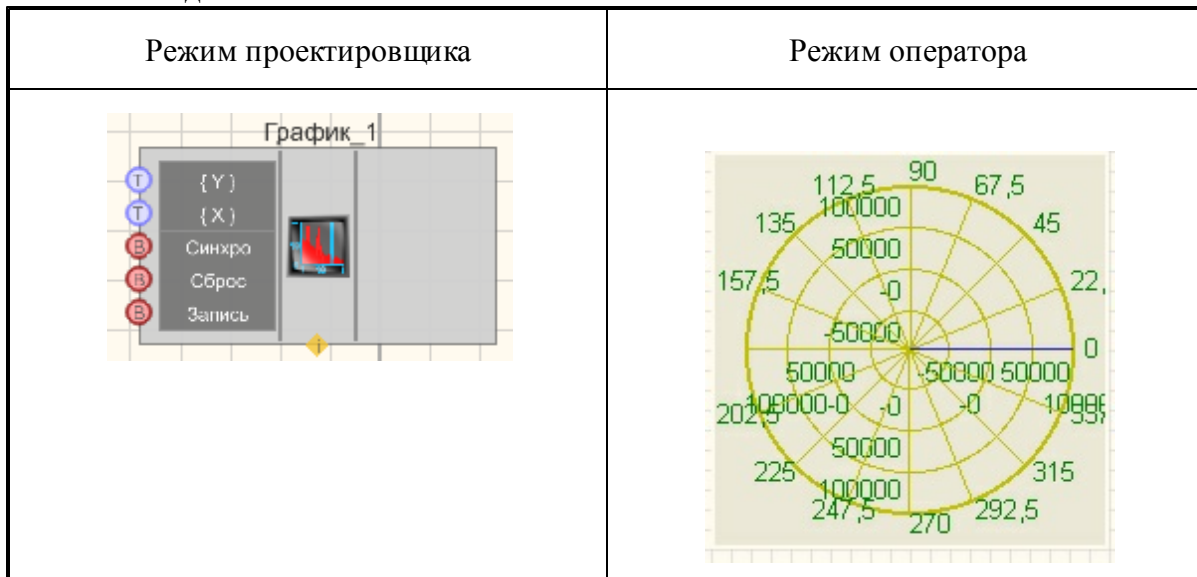


Данный пример непосредственно в ZETView

## 19.4.График в полярных координатах

<Раздел находится в разработке>

График в полярных координатах - компонент.

**Внешний вид компонента:****Параметры:**

## ➤ Входные:

- {Y} - контакт, к которому подключается массив со значениями, которые будут отображены по оси ординат.
- Синхро - контакт, к которому подключается канал, по которому будет осуществляться синхронизация по передаче данных на поле прорисовки.
- Сброс - контакт, при подаче на который импульса, происходит очистка графика.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

## ➤ Выходные

Не имеет.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

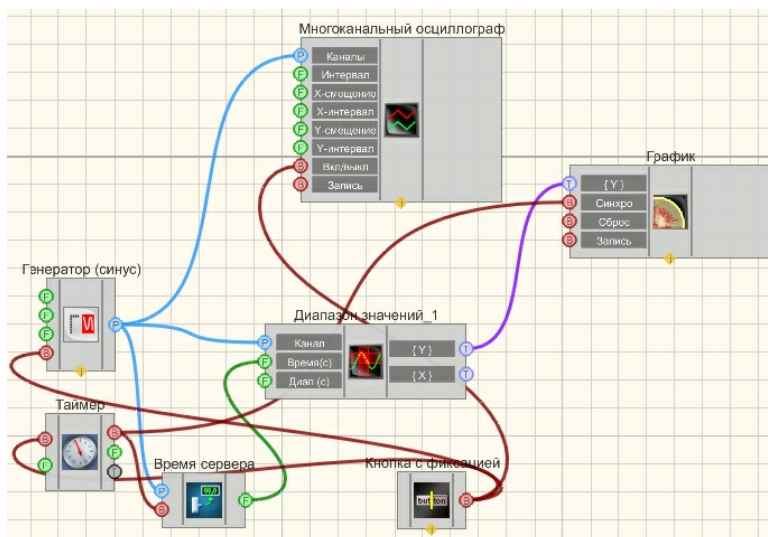
- YAutoscale (true) - включение/выключение автомасштабирования по оси Y.
- YString - установка единицы измерения по оси Y.
- Ystart (0) - начало отображения по оси Y.
- Yfinish (1) - конец отображения по оси Y.
- FileName - установка имени и расположения файла, в котором данные будут сохранены в виде отдельного файла.
- xprecision (0.01) - точность отображения значений по оси X.
- yprecision (0.01) - точность отображения значений по оси Y.
- BackColor (ece9d8) - цвет заднего фона.

- GridColor (c0c000) - цвет координатной сетки.

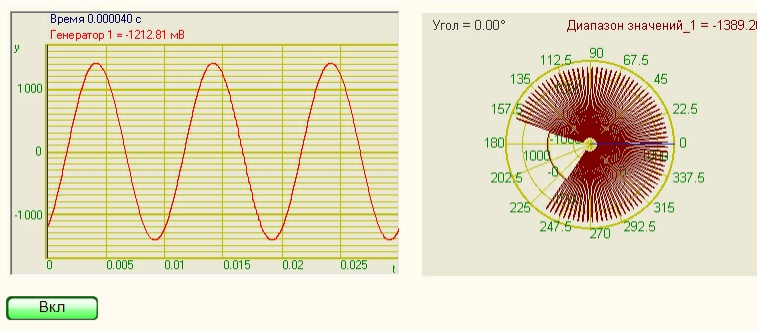


## Пример

### Проект в SCADA ZETView



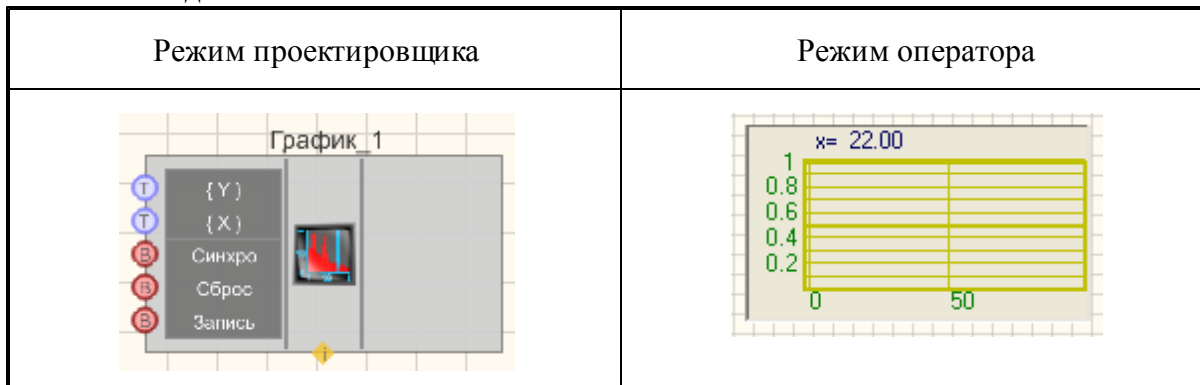
### Результат работы проекта



Данный пример непосредственно в ZETView

## 19.5.График

**График** - компонент. Предназначен для отображения численных значений на специально оформленной области.

**Внешний вид компонента:****Параметры:**

## ➤ Входные:

- {Y} - контакт, к которому подключается массив со значениями, которые будут отображены по оси ординат.
- {X} - контакт, к которому подключается массив со значениями, которые будут отображены по оси абсцисс.
- Синхро - контакт, к которому подключается канал, по которому будет осуществляться синхронизация по передаче данных на поле прорисовки.
- Сброс - контакт, при подаче на который импульса, происходит очистка графика.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- XAutoscale (true) - включение/выключение автомасштабирования по оси X.
- YAutoscale (true) - включение/выключение автомасштабирования по оси Y.
- xystate (y(n))- режим отображения графиков: [График в полярных координатах](#)<sup>[488]</sup>
- 1. y(n)
- 2. y(x) - равномерно.
- 3. y(t) - длительно.
- 4. y(x) - реально.
- XString - установка единицы измерения по оси X.
- YString - установка единицы измерения по оси Y.
- Referense (0.001) - значение для установки расчета уровней в дБ.
- Ystart (0) - начало отображения по оси Y.
- Yfinish (1) - конец отображения по оси Y.
- FileName - установка имени и расположения файла, в котором данные будут

сохранены в виде отдельного файла.

- xprecision (0.01) - точность отображения значений по оси X.
- yprecision (0.01) - точность отображения значений по оси Y.
- LineWidth (1) - установка толщины линии графика.
- BackColor (e9e9d8) - цвет заднего фона.
- GridColor (c0c000) - цвет координатной сетки.
- typeline (Горизонтальные) - тип линии графика:
  1. Горизонтальные.
  2. Ломанные.
- tureyaxis (Равномерная) - развёртка вертикальной шкалы:
  1. Равномерная.
  2. Логарифмическая.
  3. Децибельная.
- turexaxis (Равномерная) - развёртка горизонтальной шкалы:
  1. Равномерная.
  2. Логарифмическая.
  3. Децибельная.
- ColorAdjustment (False) - разрешение настройки цветов графика:
  1. True - разрешить настройку цветов графика.
  2. False - не разрешить настройку цветов графика.



### **Пример**

**Пример** использования компонента График: (находится в Примере компонента [Диапазон значений](#)<sup>[290]</sup>)

## 19.6.Карта

**Карта** - компонент. Предназначен для отображения карт и позволяет делать на ней отметки.

### **Внешний вид компонента:**

Режим проектировщика	Режим оператора



### Параметры:

#### ➤ Входные:

- Объект - контакт, для добавления нового объекта на карту.
- Очистить - контакт, при подаче на который импульса, производится очистка карты.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

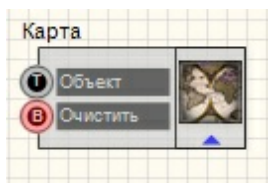
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- MapFileName - рисунок с картой.
- LeftTopLat (0) - GPS координаты левого верхнего угла (широта).
- LeftTopLong (0) - GPS координаты левого верхнего угла (долгота).
- RightBottomLat (0) - GPS координаты правого нижнего угла (широта).
- RightBottomLong (0) - GPS координаты правого нижнего угла (долгота).
- StaticObject - Статические объекты через точку с запятой. Формат - широта, долгота,радиус1,радиус2,цвет,метка.

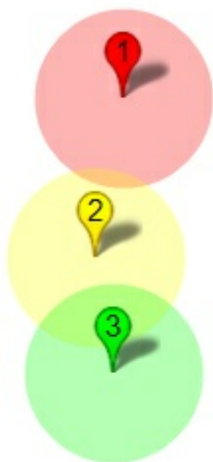


### *Пример*

### *Проект в SCADA ZETView*



*Результат работы проекта*



Данный пример непосредственно в ZETView

## 19.7. Многоканальный осциллограф

**Многоканальный осциллограф** - компонент. Является интерфейсом взаимодействия с виртуальным прибором ZETLab "Многоканальный осциллограф"

**Внешний вид компонента:**

Режим проектировщика	Режим оператора

**Параметры:**



➤ Входные:

- Каналы - контакт, к которому подключается канал, сигнал в котором требуется отобразить.
- Интервал - контакт для установления интервала отображения.
- X-смещение - контакт, при помощи которого можно сдвигать осциллограмму по оси абсцисс.
- X-интервал - контакт для установки интервала по оси X.
- Y-смещение - контакт, при помощи которого можно сдвигать осциллограмму по оси ординат.
- Y-интервал - .контакт для установки интервала по Y.
- Вкл/выкл - параметр, отвечающий за включение и выключение многоканального осциллографа.
- Запись - при подаче на данный контакт импульса происходит сохранение данных в отдельный файл (имя и местоположение файла указываются в частных свойствах компонента).

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Interval (1) - установка интервала отображения, с.
- XStart (0) - установка смещения нуля по оси X, с.
- XInterval (1) - длина отображения по оси X, с.
- YStart (0) - установка смещения нуля по оси Y (в единицах измерения).
- YInterval (1) - установка интервала отображения по оси Y (в единицах измерения).
- update (0.1) - обновление отображения (0.1 или 1 сек.).
- Synchro (false) - состояние синхронизации (включен или отключен).
- SynchroLevel (0) - уровень синхронизации (в единицах измерения).
- Activate (true) - состояние подачи сигнала (подан или отключен).
- Autoscale (false) - автомасштаб.
- freqrange (25000) - частотный диапазон, Гц.
- FileName - имя файла для записи результатов.



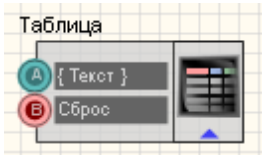
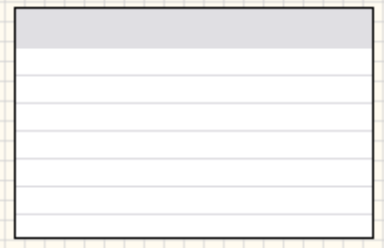
**Пример**

**Пример** использования компонента График: (находится в Примере компонента [Диапазон значений](#)<sup>[290]</sup>)

## 19.8. Таблица данных

**Таблица данных** - компонент. Предназначен для отображения текстовых значений на специально оформленной области.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- {Текст} - контакт, на который нужно подавать строковый массив. Каждый элемент будет отображаться в отдельной строке.
- Сброс - управляющий контакт, при получении импульса очищает таблицу.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

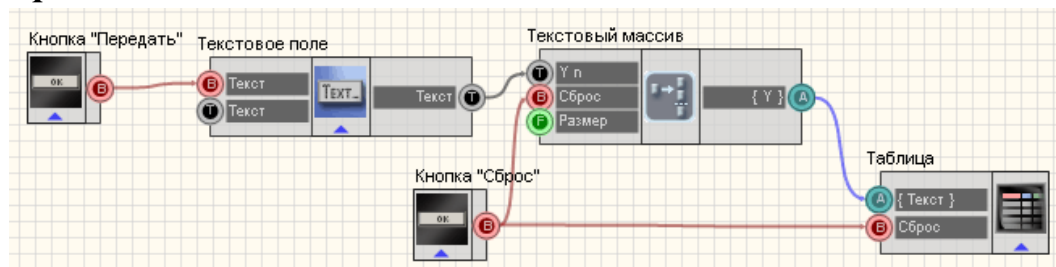
#### ➤ Частные свойства:

Не имеет.



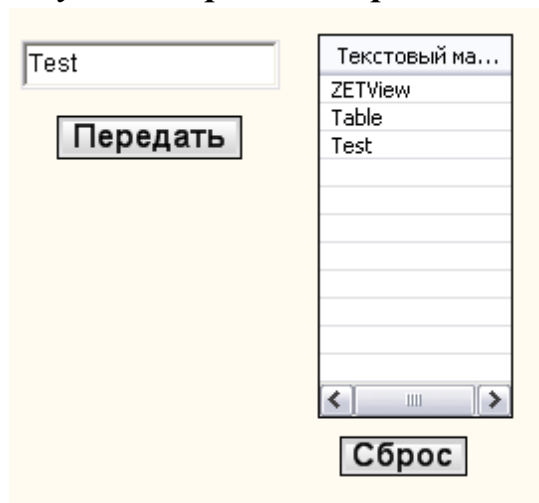
## Пример

### Проект в SCADA ZETView



В данном примере показана работа компонента [Таблица данных](#)<sup>[496]</sup>. Элементы массива вводятся пользователем в [Текстовое поле](#)<sup>[561]</sup> и отправляются на [Формировщик массивов \(текст-массив\)](#)<sup>[422]</sup> при нажатии на [Кнопки без фиксации](#)<sup>[367]</sup> массив выводится в [Таблицу данных](#)<sup>[496]</sup>.

### Результат работы проекта



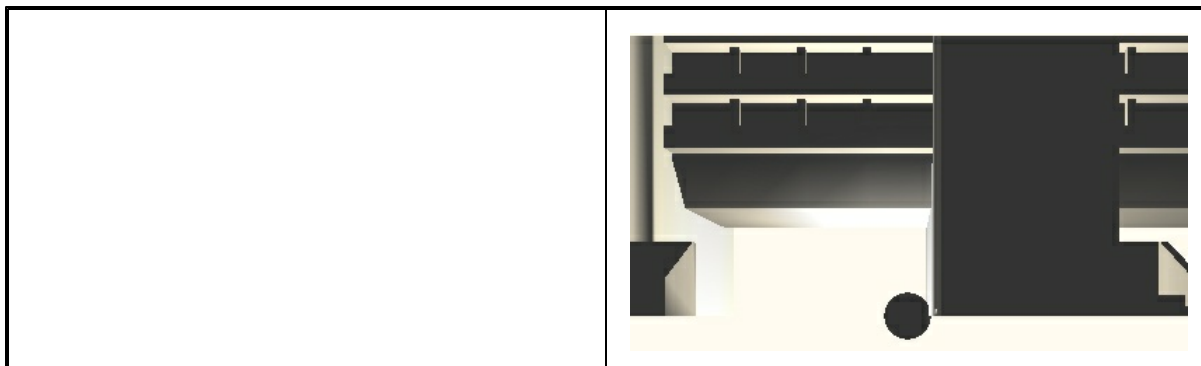
[Посмотреть пример непосредственно в ZETView](#)

## 19.9.3D-моделирование

**3D-моделирование** - компонент. Предназначен для отображения данных датчика на специально оформленной области в 3х мерном изображении.

### Внешний вид компонента:

Режим проектировщика	Режим оператора



### Параметры:

#### ➤ Входные:

- Сохранить - контакт для сохранения картинки в \*.bmp файле.
- Данные 1- данные для датчика.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ModelFileName- имя файла, в которой храниться 3D-модель.
- InfoFileName - имя файла, хранящее настройки загружаемой модели.
- SensorCount (1) - количество установленных датчиков.
- SensorVisibility (true) - видимость датчиков при запущенном проекте.
- SensorSize (2.5) -размер датчика, задается в процентах от величины, равной среднему арифметическому между линейными размерами модели.
- MaxSensorVal (1) - максимальное значение (соответствует красному цвету).
- MinSensorVal (0) - минимальное значение (соответствует синему цвету).
- SpotSizeProcent (10) - размер зоны воздействия датчика, задается в процентах к диагонали (максимальное расстояние между точками модели).
- LightCount (1) - количество установленных источников света.
- LightVisibility (true) - видимость источников света при запущенном проекте.

---

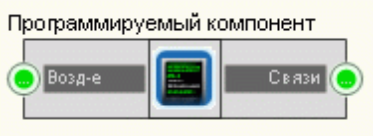
## Глава 20. Программируемая логика

### 20.1. Программируемый компонент

**Программируемый компонент** - компонент. Предназначен для расширения возможностей компонентов и реализации несложных алгоритмов посредством

использования языка программирования VBScript. В некоторых случаях позволяет минимизировать число используемых в проекте компонентов. Также позволяет вызывать методы функции компонентов. Например нажать с помощью скрипта на кнопку, выдать сообщение, поменять цвет или свойства компонента, т.е. реализовывать функции, которые не предусмотрены в системе.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Воздействие - предназначен для подключения к скрипту компонентов, воздействующих на него или данные с которых будут использоваться в программе.

##### ➤ Выходные

- Связи - предназначен для подключения к скрипту компонентов, на которые он будет воздействовать.

#### Замечание:

Компонент не обязательно должен быть присоединен к другим компонентам. В основном, соединение показывает взаимосвязь между блоками.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

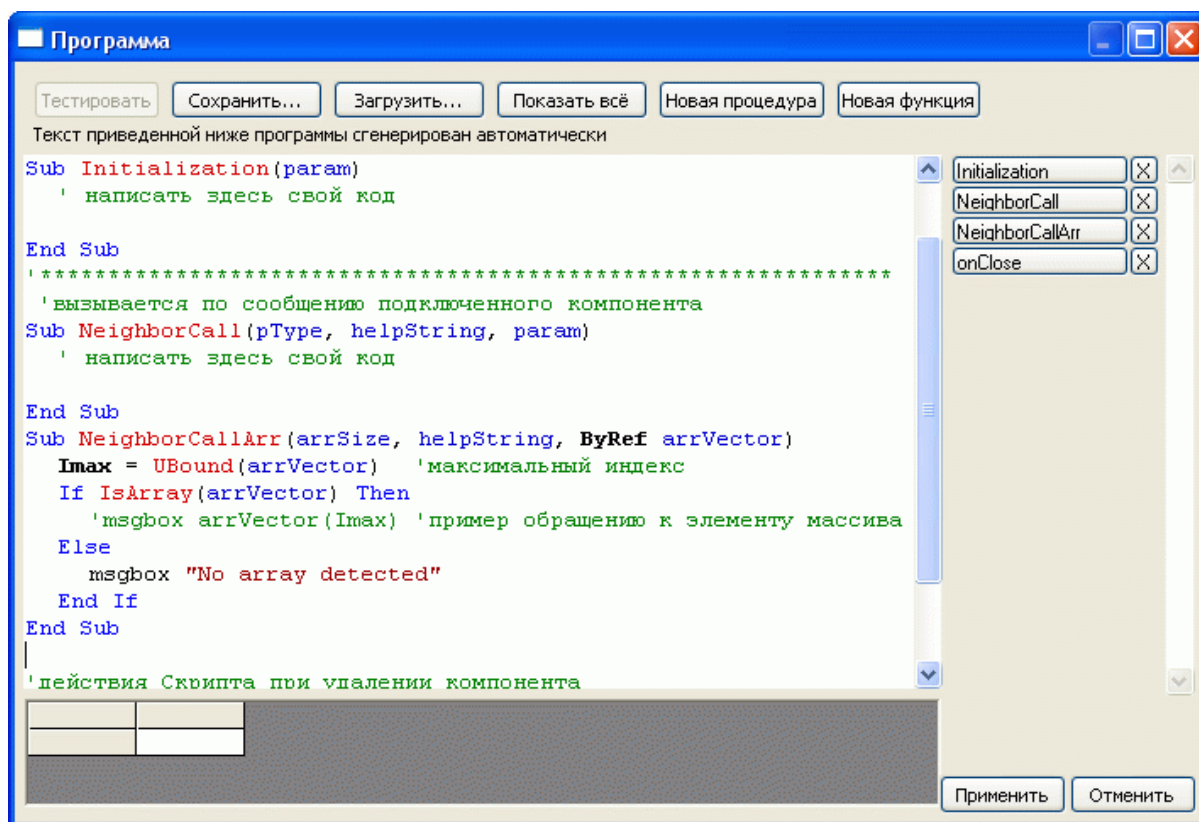
- [Приведены по данной ссылке.](#)<sup>92</sup>

##### ➤ Частные свойства:

- Не имеет.

#### Окно редактора:

При вызове контекстного меню компонента, необходимо выбрать Редактировать компонент, после чего откроется скриптовый редактор:



Окно редактора программы функционально разделено на три части: текстовый редактор (слева); перечень доступных скрипту объектов ZETView (справа); таблица обнаруженных синтаксических ошибок (снизу).

При создании текста программы по умолчанию создается следующий код:

- 1 **Sub Initialization**(param)      Данная функция вызывается при старте проекта а так же, по нажатию на кнопку Запустить проект (верхний левый угол страницы)  
     ' написать здесь свой код  
     End Sub      Предназначена для инициализации начальных параметров объектов ZETView, например, управлением состоянием управляющих компонентов (кнопок, элементов выбора и т.д).
- 2 **Sub NeighborCall**(pType, helpString, param)      Вызывается при воздействии на скрипт соседнего компонента, подключенного ко входному контакту.  
     ' написать здесь свой код      pType - тип передаваемого параметра:
  - 0 - целые числа
  - 1 - числа с плавающей запятой
  - 4 - строка (текст)
     helpString - параметр, предназначенный для идентификации источника сообщения (в случае, если на входной контакт подключены более одного компонента). Данный параметр задается в строке  
     End Sub

идентификатора.

`param` - информативный параметр. Для корректной обработки данного параметра необходимо знать значение параметра `pType`.

- Если работаем с целыми или с булевыми величинами и `pType = 0`, можно считать `param` в одну из ваших переменных напрямую.
- Если работаем с переменными с плавающей запятой, тогда для присвоения переменной значения `param` вызывается следующая функция: `This.mLtoF param,3` (где 3 - точность).

- 3 **Sub NeighborCallArr**(`arrSize`, `helpString`, **ByRef** `arrVector`) `Imax = UBound(arrVector)` `data_0` `param`
- Функция, вызываемая по приходу Event от подключенной к Unit программы.
- `size` - размер массива данных.
- `data_0` - массив данных.
- `param` - дополнительные информативные параметры.
- `Imax` - максимальный индекс
- `If IsArray(arrVector) Then`
- `msgbox arrVector(Imax)`
- 'пример обращению к элементу массива
- `Else`
- `msgbox No array detected`
- `End If`
- `End Sub`
- 4 **Function onClose**(`param`)
- Функция, вызываемая при закрытии программы, а так же при остановке ее выполнения.
- `'текст вашей программы`
- `End Function`

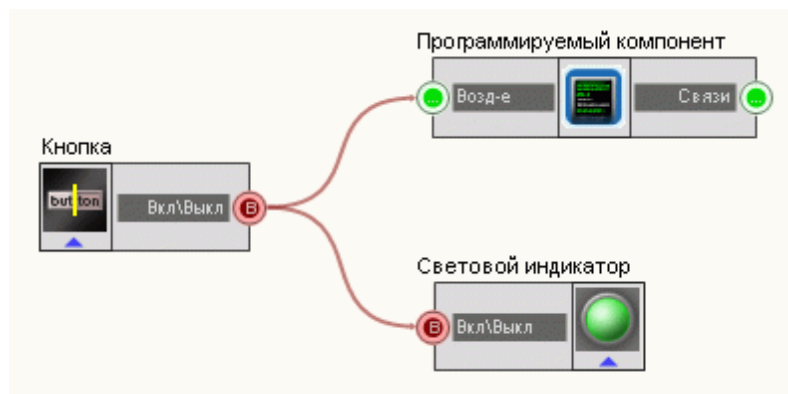
Синтаксис языка VBScript, описание функций, а так же, доступ к стандартным объектам Windows и прочие детали приведены на официальном сайте Microsoft:

- [http://msdn.microsoft.com/en-us/library/t0aew7h6\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/t0aew7h6(VS.85).aspx)
- [http://msdn.microsoft.com/ru-ru/library/bx9ceb2w\(en-us,VS.85\).aspx](http://msdn.microsoft.com/ru-ru/library/bx9ceb2w(en-us,VS.85).aspx)



### Пример 1

### Проект в SCADA ZETView



В этой схеме компонент **Программируемый компонент** служит для инициализации цветов в компонентах [Кнопка без фиксации](#)<sup>[367]</sup> и [Световой индикатор](#)<sup>[356]</sup>. В данном случае была упрощена реализация задания цветов для блоков с помощью **Программируемый компонент**. При нажатии на [Кнопку без фиксации](#)<sup>[367]</sup> появляется диалоговое окно, в котором сообщается какое значение параметра было передано [Световому индикатору](#)<sup>[356]</sup> с компонента [Кнопка без фиксации](#)<sup>[367]</sup>.

```

'***** точка входа в программу *****
'вызывается по сообщению при старте основной программы
'CallMessageName - комментарий к передаваемому параметру
Sub Initialization(param)
    кнопка.OnBackColor = &H808000
    кнопка.OffBackColor = &HCD5C5C
    световой_индикатор.onColor = &H90EE90
    световой_индикатор.offColor = &H800000

End Sub
'*****
'вызывается по сообщению подключенного компонента
Sub NeighborCall(pType, helpString, param)
    msgbox param

End Sub
Sub NeighborCallArr(arrSize, helpString, ByRef arrVector)
    Imax = UBound(arrVector) 'максимальный индекс
    If IsArray(arrVector) Then
        'msgbox arrVector(Imax) 'пример обращению к элементу массива
    Else
        msgbox No array detected
    End If
End Sub

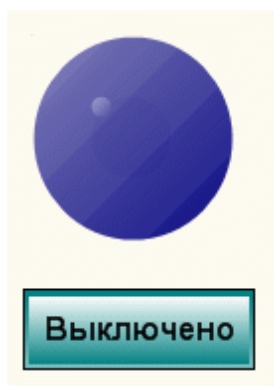
'действия Скрипта при удалении компонента

```



```
'или иной команде, завершающей работу скрипта  
Function onClose(param)  
    'текст вашей программы  
End Function
```

### *Результат работы проекта*



Данный пример непосредственно в ZETView



### *Пример 2*

#### **Вопрос.**

При работе с использованием программируемого компонента возникает следующая проблема: т.к. его процедуры и функции написаны на VBScript, то при окончании скрипта значения всех его глобальных переменных теряются. Другими словами, если в скрипте используется процедура, которой нужны данные, полученные в результате предыдущего вызова скрипта, лежащие в глобальных переменных, то она их не получит. Нет эффективного способа сохранять значения глобальных переменных скрипта между итерациями. Как решить эту задачу?

**Ответ.** В ниже указанном примере указано, как нужно работать с

глобальными переменными.

**Примечание.**

- 1) К компонентам в VBScript можно обращаться не только по HelpString, но и по Caption, заменив пробелы символами подчеркивания «\_».
- 2) Написать свой компонент для ZETView, который можно добавить в список Перечень компонентов виртуальных приборов, нельзя.

Данный пример непосредственно в ZETView

## 20.2.Список переменных к сценарию

**Список переменных к сценарию** - компонент. Предназначен для реализации несложных алгоритмов посредством определенных команд. Редактор выполнен в виде таблицы, в которой наглядно, в прямом порядке записаны исполняемые команды, формирующие последовательность действий. В некоторых случаях позволяет минимизировать число используемых в проекте компонентов.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Импульс - при поступлении импульса на этот контакт, значения переменных из таблицы передаются на контакт список.
- Данные - количество циклов повторения исполнения сценария.

➤ Выходные

- Список - с этого контакта идет передача значений переменных на другие компоненты.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

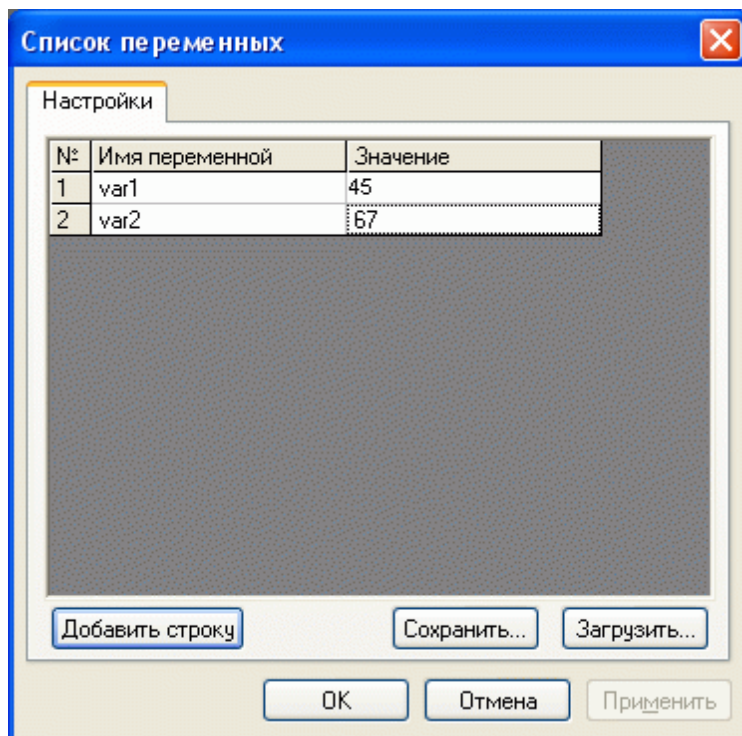
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства:

- Не имеет.

### Окно редактора:

При вызове контекстного меню компонента, необходимо выбрать Редактировать компонент, после чего откроется редактор переменных:



Окно редактора Список переменных разделено на три колонки. Первая колонка - номер переменной. Вторая Имя переменной - название переменной. Третья колонка Значение - значение переменной.



### **Пример**

**Пример** использования Списка переменных: (находится в Примере компонента [Сценарий](#)<sup>505</sup>)

## 20.3.Сценарий

**Сценарий** - компонент. Предназначен для реализации несложных алгоритмов посредством определенных команд. Редактор выполнен в виде таблицы, в которой наглядно, в прямом порядке записаны исполняемые команды, формирующие

последовательность действий. В некоторых случаях позволяет минимизировать число используемых в проекте компонентов.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Выполнить - начинается выполнение сценария.
- Циклы - количество циклов повторения исполнения сценария.
- Время - установка численного значения паузы между двумя циклами, мс.
- Список - подключение списка переменных для этого сценария.

##### ➤ Выходные

- Инфо - информация о ходе выполнения сценария..
- Состояние - конец выполнения сценария.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

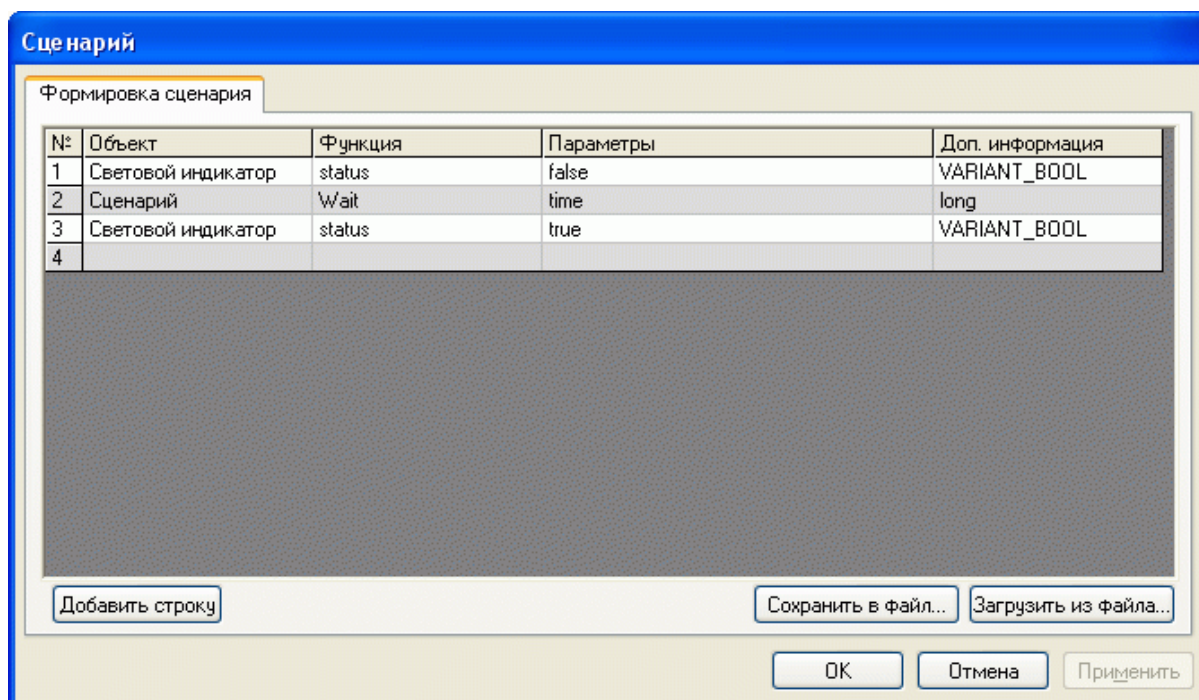
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- CycleQuantity (1) - количество циклов выполнения сценария.
- CycleWaitTime (0) - время ожидания между циклами, с.
- HappyEndString (Сценарий успешно выполнен!) - строка, значение которой будет выведено в случае успешного выполнения сценария.
- BadEndString (Выполнение сценария прервано!)- строка, которая будет выведена в случае прерывания сценария.
- ProcessString (Сценарий выполняется...) - строка, которая выводится во время выполнения сценария.
- StopPosition (500) - номер строки на которой будет остановлено выполнение сценария.

#### Окно редактора:

При вызове контекстного меню компонента, необходимо выбрать Редактировать компонент, после чего откроется скриптовый редактор в форме:

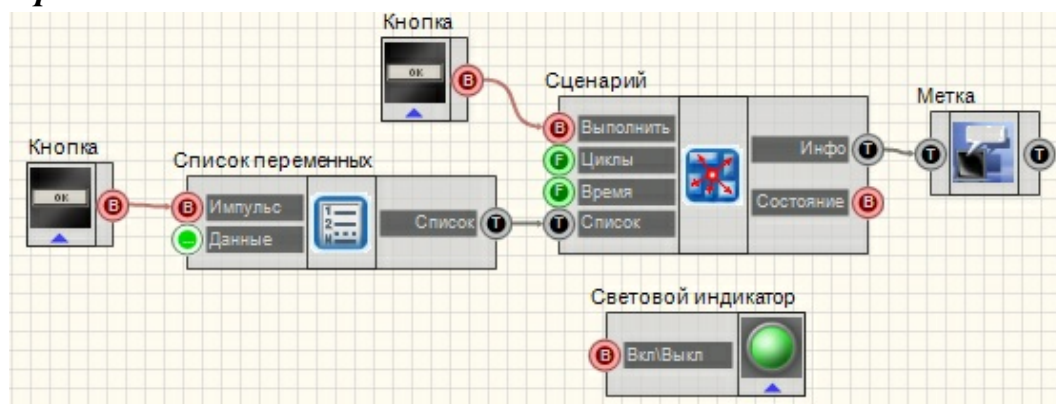


Окно редактора разделено на четыре колонки. Первая колонка - номер исполняемой команды. Вторая Объект - название объекта, с которым происходит работа. Третья колонка Параметры - значение параметра, с которым происходят изменение. Четвертая колонка Доп. информация - дополнительная информация о свойстве объекта.



## Пример

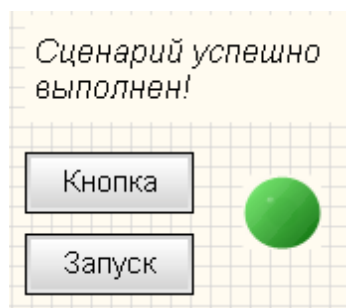
### Проект в SCADA ZETView



В этой схеме компонент Сценарий служит для реализации

срабатывания (включения-выключения) [Светового индикатора](#)<sup>[356]</sup>. [Список переменных](#)<sup>[504]</sup> нужен для задания таблицы значений переменных. [Кнопка без фиксации](#)<sup>[367]</sup>, подключенная к [Списку переменных](#)<sup>[504]</sup>, подает импульс на [Сценарий](#)<sup>[505]</sup> после чего идет передача значений переменных в компонент [Сценарий](#)<sup>[505]</sup>. [Метка](#)<sup>[554]</sup> выводит на экран текстовое сообщение о состоянии выполнения сценария. Оставшийся компонент [Кнопка](#)<sup>[367]</sup> служит для подачи импульса для начала исполнения сценария.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 20.4. Цикл for

**Цикл for** - компонент. Предназначен для реализации цикла с счетчиком, т.е. для повторения выполнения части проекта определенное число раз.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Значение - начальное значение переменной цикла.
- Импульс - импульс окончания такта цикла.
- Импульс - импульс прерывания цикла.

#### ➤ Выходные

- Выход - выходная переменная в тело цикла.

- Импульс - импульс об окончании цикла.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

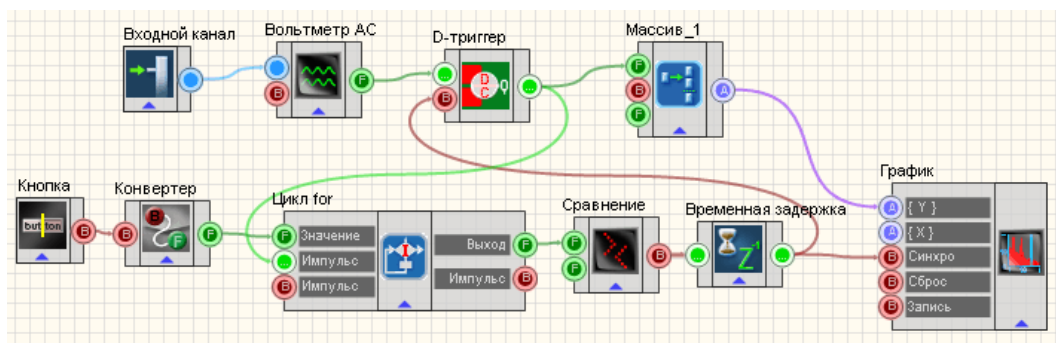
- Quantity (1) - установка количества циклов.
- Increment (0) - значение выходного инкремента.
- Delay (0) - задержка выполнения следующего такта цикла, с.



### Пример



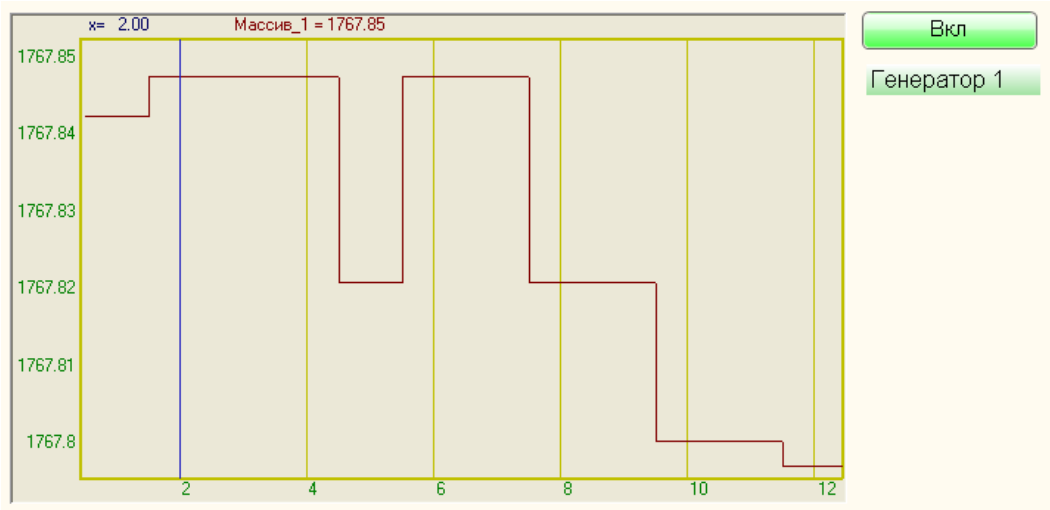
*Для работы примера необходимо требуется подключенное ZET-устройство, содержащее ЦАП.*



В данном проекте показана работа компонента "Цикл for". Данные с генератора поступают на компонент "Вольтметр АС". Измеренные значения вольтметра поступают на компонент "Формировщик массива (число-массив)" после чего данные выводятся на график.

Значения с вольтметра поступают в момент срабатывания D-триггера. Триггер управляется компонентом "Цикл for". При каждой итерации цикла, поступает импульс на синхровход триггера. Поэтому количество срабатываний триггера зависит от количества итераций, заданных в настройках компонента "Цикл for". При рассмотрении схемы можно выделить блок измерения (сверху) и блок цикла (снизу). Вся схема начинает работать при включении блока цикла с помощью кнопки с фиксацией. Кнопка подключена к компоненту конвертер (Bool-число), который преобразует логический тип данных в численный. То есть при настройках по умолчанию и высоком логическом уровне на входе, конвертер выдаст 1. При низком -0. Затем число 1 поступает на вход "Значение" компонента "Цикл

for", тем самым задавая начальное значение счетчика цикла и запуская работу компонента. Выходное значение цикла for сравнивается с отрицательным значением на компоненте "Сравнение неравенство". Это сделано для того, чтобы просто конвертировать численное выходное значение с компонента "цикл for" в импульс, который будет управлять синхровходом D-триггера. Поэтому на компоненте "Сравнение" задано условие, которое будет всегда истинно, соответственно импульсы будут поступать на D-триггер при каждой итерации цикла. Компонент Временная задержка позволяет управлять временем срабатывания триггера.



Данный пример непосредственно в ZETView

## 20.5. Цикл while

<Раздел находится в разработке>

**Цикл while** - компонент. Предназначен для реализации цикла, т.е. для повторения выполнения части проекта определенное число раз.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет



**Параметры:**

## ➤ Входные:

- Импульс - импульс начало цикла.
- Значение - выходное значение переменной цикла
- Импульс - импульс прерывания цикла.

## ➤ Выходные

- Импульс - импульс начала следующего такта цикла
- Импульс - импульс об окончании цикла

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Value (0) - порог окончания цикла.
- Delay (0) - задержка выполнения следующего такта цикла, с.
- stop (по превышению порога) - окончание цикла:
  1. по превышению порога.
  2. по принижению порога.

## Глава 21. Поточковая обработка

### 21.1. Арифмометр

**Арифмометр** - компонент. Предназначен для совершения математических действий с двумя каналами в реальном времени.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Канал 1 - вывод, на который поступает поток данных канала №1.

- Канал 2 - вывод, на который поступает поток данных канала №2.
- Выходные:
- Канал - вывод, с которого выводится результирующий поток данных (постоянно и в реальном времени).

**Настраиваемые свойства:**

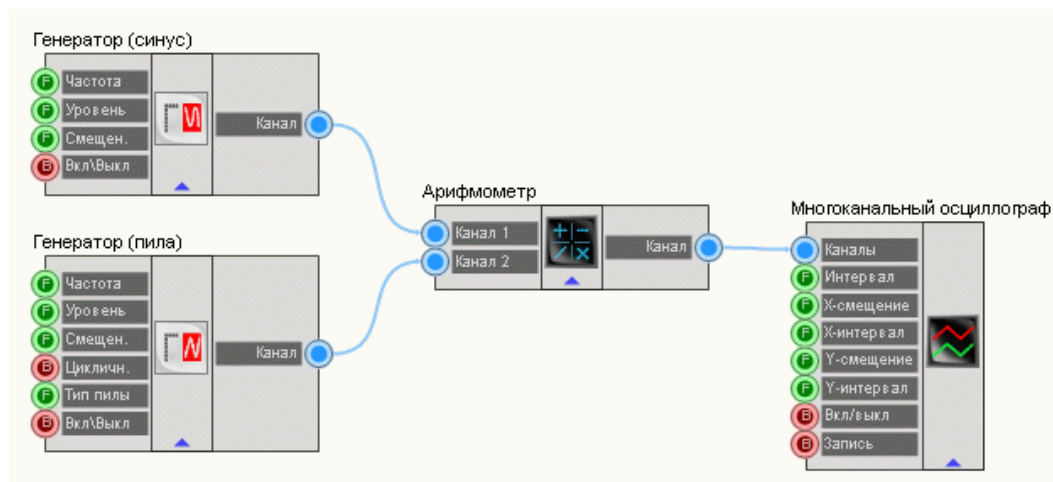
➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

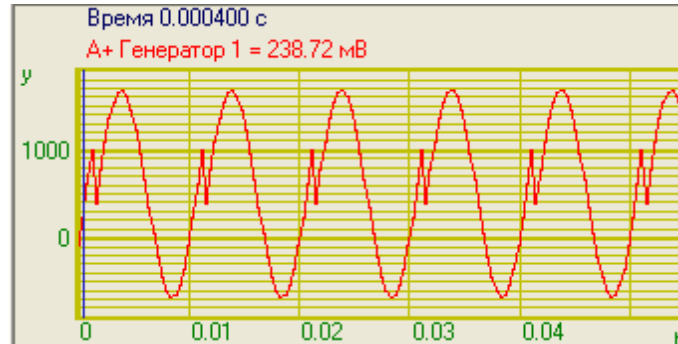
- operation (Сложение) - установка типа арифметической операции проводимой с входными каналами:
  1. Сложение - суммирование значений входных каналов.
  2. Вычитание - разность значений первого и второго каналов (значения канала №2 вычитаются из канала №1).
  3. Умножение - произведение значений каналов.
  4. Деление - частное первого и второго каналов (значения канала №1 делятся на значения канала №2).
  5. Максимум - вывод только максимальных значений в каждый момент времени работы.
  6. Минимум - вывод только минимальных значений в каждый момент времени работы.
  7. Среднее арифметическое - вывод среднего значения двух сигналов (сумма значений канала №1 и канала №2 деленная на 2).
  8. Модуль - вывод модуля сигнала (т.е. корня квадратного из суммы квадратов входных значений каналов).
  9. Среднее геометрическое - вывод среднегеометрического значения двух сигналов (корень квадратный из произведения значений канала №1 и канала №2).
- MultConstant (1) - установка константы для умножения.
- AddConstant (0) - установка константы для сложения.

***Пример******Проект в SCADA ZETView***



В этой схеме компонент Арифмометр служит для сложения двух независимых сигналов в один. [Многоканальный осциллограф](#)<sup>[494]</sup> необходим для графического отображения результирующего сигнала. [Синусоидальный сигнал](#)<sup>[247]</sup> и [Пилообразный сигнал](#)<sup>[239]</sup> используются для получения двух сигналов разного типа.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## 21.2.Компаратор

**Компаратор** - компонент. Предназначен для сравнения двух входных значений сигнала, и в зависимости от того, превышает ли уровень на первом входе уровень на втором на выходе, устанавливается высокий или низкий логический уровень на выходе.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал 1 - вывод, на который поступает поток данных канала №1.
- Канал 2 - вывод, на который поступает поток данных канала №2.

#### ➤ Выходные:

- Импульс - вывод, на котором выставляется логическая единица, если значение сигнала в канале №1 превышает значение в канале №2, и ноль, если наоборот.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

#### Частные свойства:

Не имеет.



### *Математическое описание*

Очень часто бывает нужно установить, какой из двух сигналов больше, или определить, когда сигнал достигнет заданного значения. Например, при генерации треугольных колебаний через конденсатор пропускают положительный или отрицательный ток, полярность тока изменяется в тот момент, когда амплитуда достигает заданного пикового значения. Другим примером служит цифровой вольтметр. Для того, чтобы преобразовать напряжение в код, на один из входов компаратора подают неизвестное напряжение, а на другой – линейно-нарастающее напряжение (конденсатор+источник тока). Цифровой счетчик подсчитывает периоды генератора, пока линейно-нарастающее напряжение меньше, чем неизвестное; в момент равенства амплитуд производится считывание результата, полученного на счетчике. Результат пропорционален входному напряжению. Такое преобразование называют интегрированием с одним углом наклона; в более сложных приборах используют интегрирование с двумя углами наклона.


Простейшим компаратором является дифференциальный усилитель с большим коэффициентом усиления, построенный на основе транзисторов или операционных усилителей. В зависимости от знака разности входных напряжений операционный усилитель оказывается в положительном или отрицательном насыщении. Коэффициент усиления по напряжению обычно превышает 100000, поэтому, для того, чтобы выход усилителя не насыщался, напряжение на входах должно быть равно долям милливольт. Для компараторов обычно не используют термин “скорость нарастания”, вместо этого говорят о задержке распространения относительно сигнала, заданного на вход.

## 21.3.Пиковый обнаружитель

<Раздел находится в разработке>

**Пиковый обнаружитель** - компонент. Предназначен для поиска максимального и минимального значений сигнала на заданном интервале.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Канал - вывод, на который поступает поток данных канала.
- Время - время сервера, с
- Задержка - задержка поиска пикового значения.
- Интервал - интервал поиска пикового значения, с.

➤ Выходные

- Минимум - на выходе получаем минимальное пиковое значение.
- t мин. - время минимального пикового значения, с.
- Максимум - на выходе получаем максимальное пиковое значение.

- t макс. - время максимального пикового значения, с.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Maximum (true) - определение максимального пикового значения.
- Minimum (true) - определение минимального пикового значения.
- Delay (0) - значение задержки пикового экстремума, с.
- Interval (0.1) - значение интервала поиска экстремума, с.

## 21.4. Синхронизатор (1 канал)

<Раздел находится в разработке>

**Синхронизатор (1 канал)** - компонент. Предназначен для определения момента времени одного канала, когда уровень анализируемого сигнала достигнет заданного значения.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Канал - вывод, на который поступает поток данных канала.
- Ц.полосы - центральное значение полосы.
- Ш.полосы - ширина полосы.
- Вкл/выкл - Вкл \ Выкл - включение\выключение канала синхронизатора.

##### ➤ Выходные

- Синхро - импульс синхронизации.
- Время - время импульса синхронизации.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Level (0) - уровень синхронизации.
- LevelWidth (0.05) - ширина уровня синхронизации.
- frontype (Восходящий) - тип фронта синхронизации:
  1. Восходящий;
  2. Нисходящий;
  3. Оба.
- Active (true) - состояние. Включение\выключение канала синхронизатора
- Module (false) - обработка данных по модулю.
- Unsensetivity (false) - отсутствие чувствительности в течение определенного времени после срабатывания.
- UnsensetivityTime (0) - время отсутствия чувствительности после срабатывания.

## 21.5. Синхронизатор (4 канала)

<Раздел находится в разработке>

**Синхронизатор (4 канала)** - компонент. Предназначен для определения момента времени четырёх каналов, когда уровень анализируемого сигнала достигнет заданного значения.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Канал - вывод, на который поступает поток данных канала.
- Ц.полосы - центральное значение полосы.
- Ш.полосы - ширина полосы.
- Вкл/выкл - Вкл \ Выкл - включение\выключение канала синхронизатора.

## ➤ Выходные

- Импульс - на выходе получаем импульс (Да - больше, Нет- меньше)
- Время - время импульса первого канала.
- Время - время импульса второго канала.
- Время - время импульса третьего канала.
- Время - время импульса четвертого канала.
- Данные - массив данных
- t данных - размерность.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ABSInputData (false) - входные данные брать по модулю.
- JustFirstValue (false) - синхронизироваться только по первому значению.

## Глава 22. Протоколы обмена и шины данных

### 22.1. Интерфейс RS-232

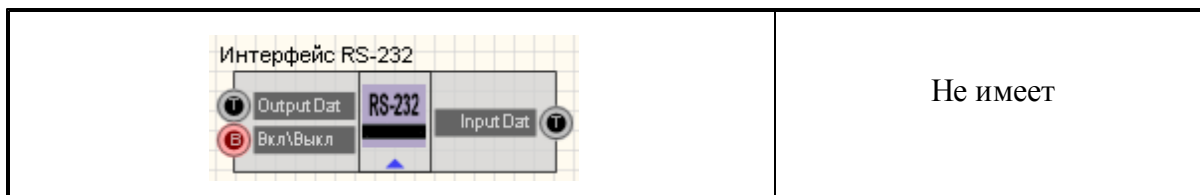
<Раздел находится в разработке>

**Интерфейс RS-232** - компонент. Предназначен для подключения к компьютеру стандартных внешних устройств (принтера, сканера, модема, мыши и др.), а также для связи компьютеров между собой.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------





### Параметры:

#### ➤ Входные:

- OutputDat - отправляемые данные.
- Вкл/Выкл - начало или окончание работы по интерфейсу RS-232.

#### ➤ Выходные

- InputDat - массив принятых значений.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства:

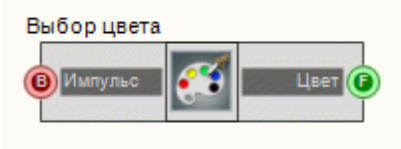
- PortNumber (7) - номер последовательного (COM) порта для работы с источником питания
- bode (9600) - скорость передачи данных.
- parity (N) - чётность (E, M, N, 0, S).
- data (8) - биты данных (4, 5, 6, 7, 8).
- stopbit (1) - количество стоповых битов.

## Глава 23.Разное

### 23.1.Выбор цвета

**Выбор цвета** - компонент. Служит для выбора и вывода кода цвета создания доступа к частным свойствам компонентов, дублируя все эти параметры в виде отдельного модуля на рабочем поле в режиме разработчика.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - входной контакт, при подаче на который импульса, появляется палитра цвета, в которой пользователь может выбрать цвет, который будет передаваться с выходного контакта.

#### ➤ Выходные

- Цвет - выходной контакт, с которого передается число определяющий цвет в формате RGB.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

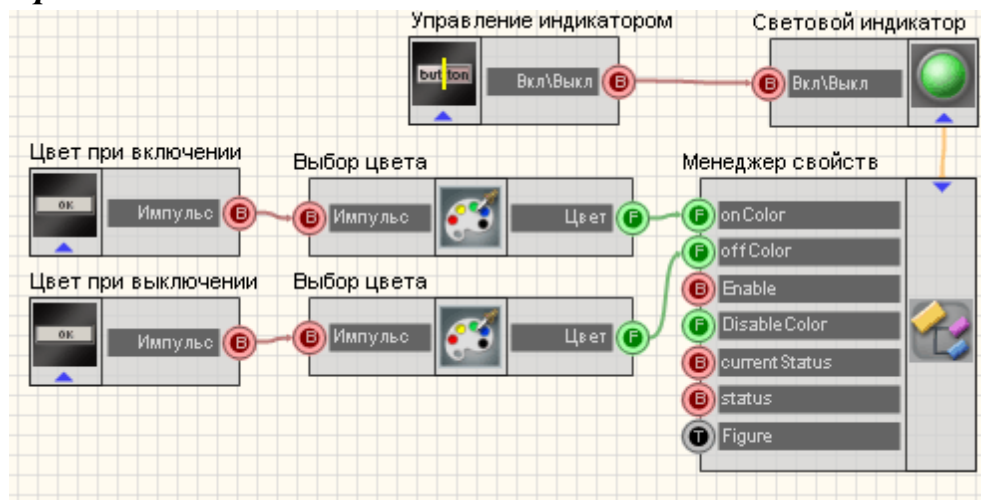
#### ➤ Частные свойства:

Не имеет.



## Пример 1

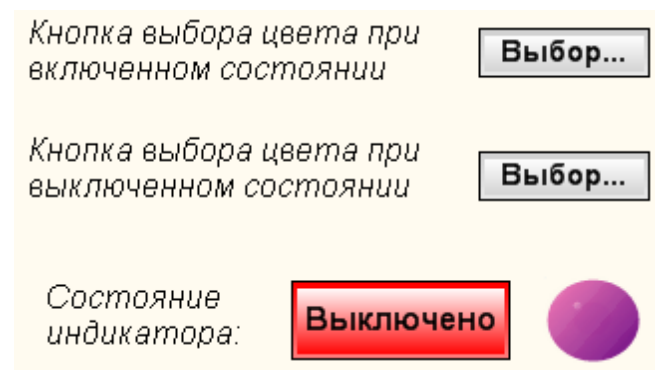
### Проект в SCADA ZETView:



В данной схеме компонент [Менеджер свойств](#)<sup>[534]</sup> открывает доступ

для других модулей к частным свойствам. [Световой индикатор](#)<sup>[356]</sup> компонент служит для задания цвета индикатора в разных режимах. [Кнопки без фиксации](#)<sup>[367]</sup> нужны для вызова цветовой палитры. [Кнопка с фиксацией](#)<sup>[366]</sup> служит для переключения состояния индикатора. [Световой индикатор](#)<sup>[356]</sup> служит для графического отображения изменения цвета.

### ***Результат работы проекта***

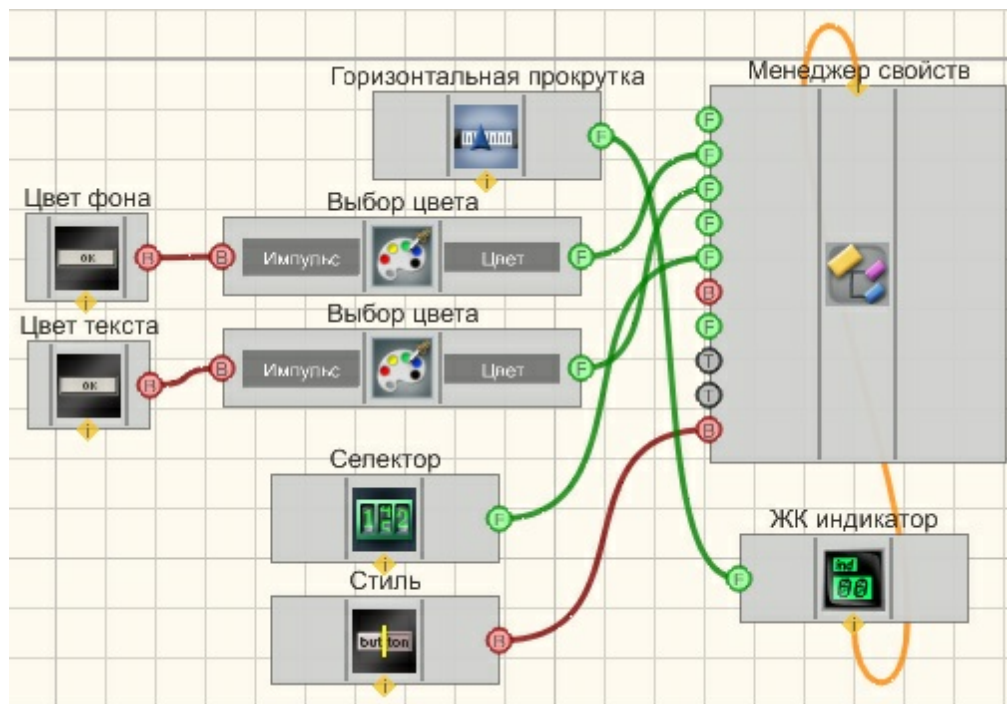


Данный пример непосредственно в ZETView



### ***Пример 2***

***Проект в SCADA ZETView:***



*Результат работы проекта*



Данный пример непосредственно в ZETView

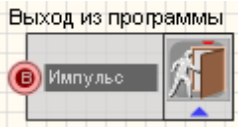

## 23.2.Выход из программы

**Выход из программы** - компонент. Предназначен для выхода из программы ZETView при получении управляющего импульса на вход, либо при нажатии на кнопку в режиме

оператора.

Если в проекте были произведены изменения, то перед выходом ZETView спросит, сохранять ли изменения, и завершит работу после того, как пользователь выберет ответ.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

#### Параметры:

##### ➤ Входные:

- Импульс - контакт, на который подается управляющий импульс для выхода из программы.

##### ➤ Выходные:

Не имеет.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

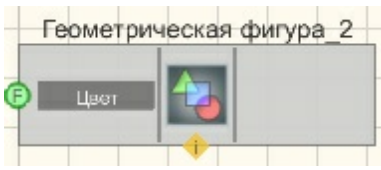
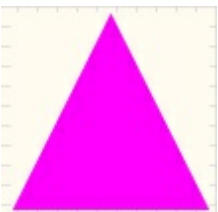
##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- TextSize (12) - размер текста надписи на кнопке в режиме оператора.
- textStyle (FontStyleBold) - стиль текста надписи на кнопке в режиме оператора.
- Enable (true) - доступность компонента.
- ButtonCaption (Выход) - текст надписи на кнопке. "Выход" - по умолчанию.

## 23.3. Геометрическая фигура

**Геометрическая фигура** - компонент. Предназначен для выбора формы: прямоугольник (круг, треугольник), цвета и с управляемыми координатами (X,Y).

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Цвет - присваивается цвет геометрической фигуры.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

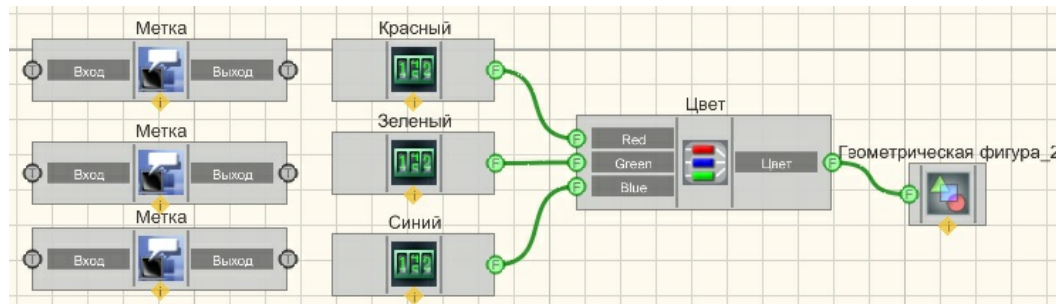
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Color (000000) - выбираем цвет геометрической фигуры.
- Figuretype (Круг) - тип геометрической фигуры:
  1. Круг.
  2. Прямоугольник.
  3. Треугольник.



## Пример 1

### Проект в SCADA ZETView



В этой схеме компонент [Геометрическая фигура](#)<sup>[523]</sup> предназначен для изменений цвета значения которых поступают с [Селектора №1](#)<sup>[213]</sup>, [Селектора №2](#)<sup>[213]</sup>, [Селектора №3](#)<sup>[213]</sup>

## Результат работы проекта

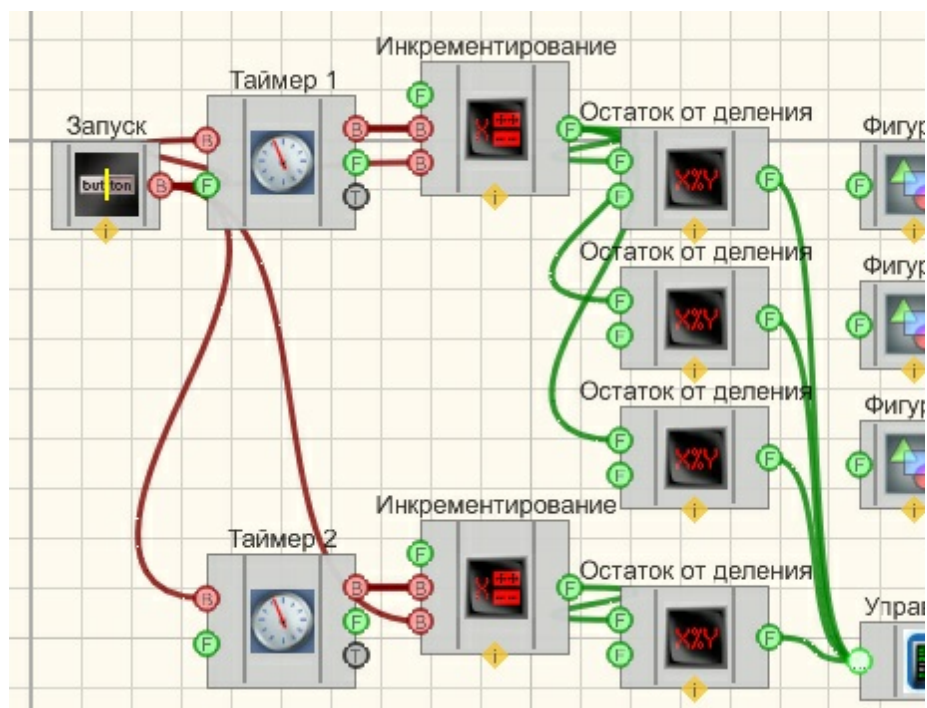


Данный пример непосредственно в ZETView



## Пример 2

### Проект в SCADA ZETView



В этой схеме компонент Геометрическая фигура предназначен с управляемыми координатами (X,Y).

### *Результат работы проекта*

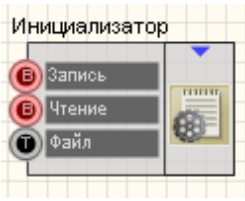


Данный пример непосредственно в ZETView

## 23.4.Инициализатор

**Инициализатор** - компонент. Предназначен для инициализации частных свойств компонента через внешний текстовый файл.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- **Запись** - при подаче импульса на этот контакт происходит запись всех текущих частных свойств подключенных к нему компонентов во внешний текстовый файл.
- **Чтение** - при подаче импульса на этот контакт происходит инициализация всех считанных из файла частных свойств.
- **Файл** - имя файла, из которого будут браться данные сигнала для инициализации.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>



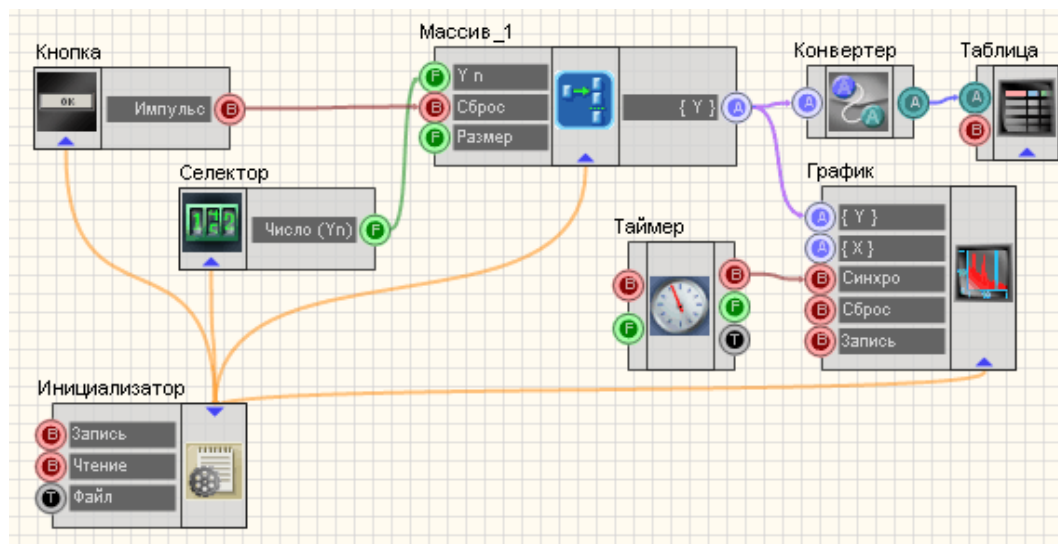
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IniFileName - выбираем место и имя файла, в котором будут записаны значения для инициализации.
- inittype (Комбинированно) - тип инициализации:
  1. Только при загрузке.
  2. Только по запросу.
  3. Комбинированно.
- WriteFile\_OnStop (true) - включение/отключение сохранения значений в файл при остановке проекта.
- InitComp\_OnRun (true) - включение/отключение инициализации значений при старте проекта.



## Пример

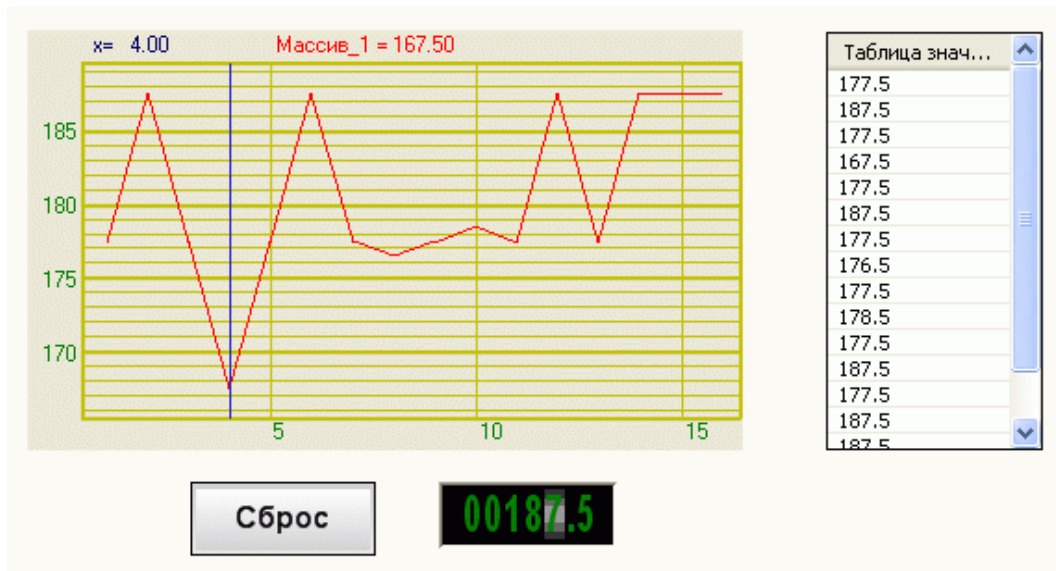
### Проект в SCADA ZETView



В этой схеме компонент [Формировщик массивов \(число-массив\)](#)<sup>[424]</sup> предназначен для заполнения массива числами, которые поступают с [Селектора №1](#)<sup>[213]</sup>. [Кнопка без фиксации](#)<sup>[367]</sup> используется для обнуления массива. [Таймер](#)<sup>[183]</sup> служит для подачи синхроимпульсов на [График](#)<sup>[490]</sup>. [Конвертер \(числовой массив - строковый массив\)](#)<sup>[378]</sup> нужен для преобразования типа данных в форму, совместимую с [Таблицей данных](#)<sup>[496]</sup> (этот компонент позволяет отображать данные массива в форме таблицы). Инициализатор служит для сохранения частных свойств компонентов при выходе, а также инициализации записанных значений этих же свойств при

последующем запуске.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 23.5.Концентратор

**Концентратор** - компонент. Предназначен исключительно для удобства расположения компонентов в режиме проектировщика. Позволяет менять местами вход и выход, а также переносить их на одну сторону. Не производит никаких действий с полученными данными.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет.

**Параметры:**

## ➤ Входные:

- Вход - входной контакт.

## ➤ Выходные

- Выход - выходной контакт.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>


## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- input (слева) - положение входного контакта (слева/справа).
- output (справа) - положение выходного контакта (слева/справа).
- 

## 23.6. Межстраничная связь

**Межстраничная связь** - компонент. Предназначен для связи между компонентами, расположенными на разных страницах проекта.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Вход (доступен только в режиме передачи) - входной контакт, на который могут передаваться данные любого типа.

## ➤ Выходные

- Выход (доступен только в режиме приема) - выходной контакт, с которого передаются данные любого типа, полученные со входа.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

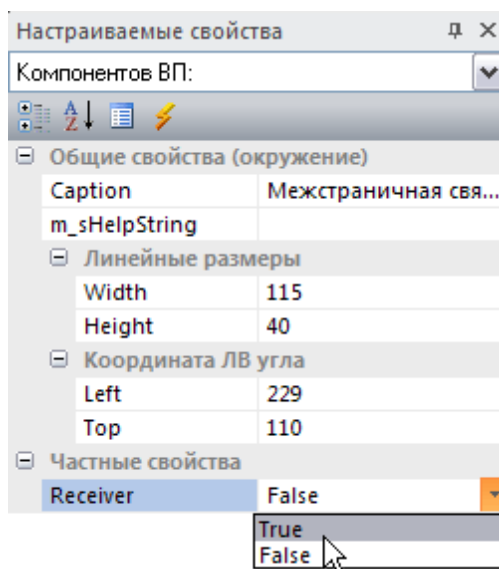
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

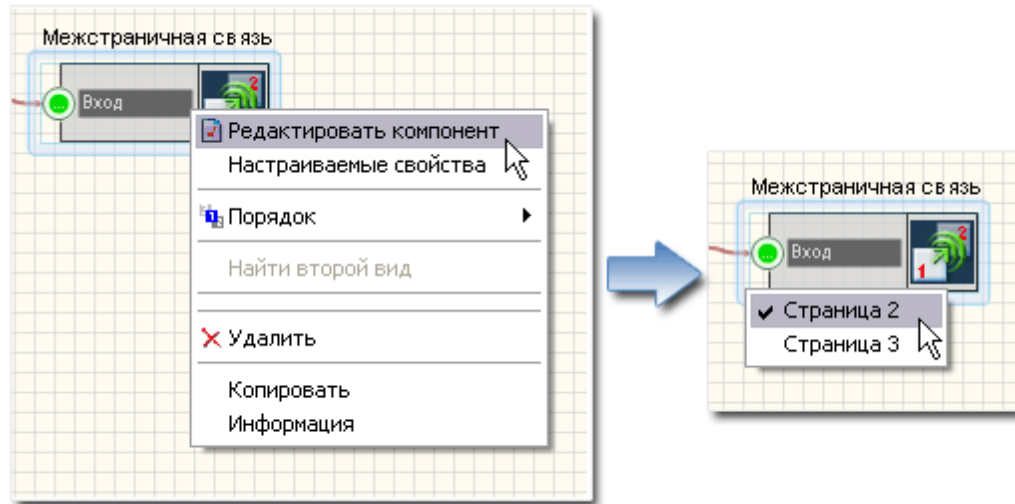
- Receiver (false) - принимает значения True/False, переводит компонент в режим передачи или приема. В зависимости от установленного режима у компонента меняется внешний вид и расположение контактов: компонент имеет только входной контакт если значение Receiver=True, и только выходной контакт, если Receiver=False.

Компонент имеет особенность настройки.

Сначала необходимо выставить свойство Receiver=True, на тех компонентах "Межстраничная связь", которые будут играть роль принимающих компонентов:

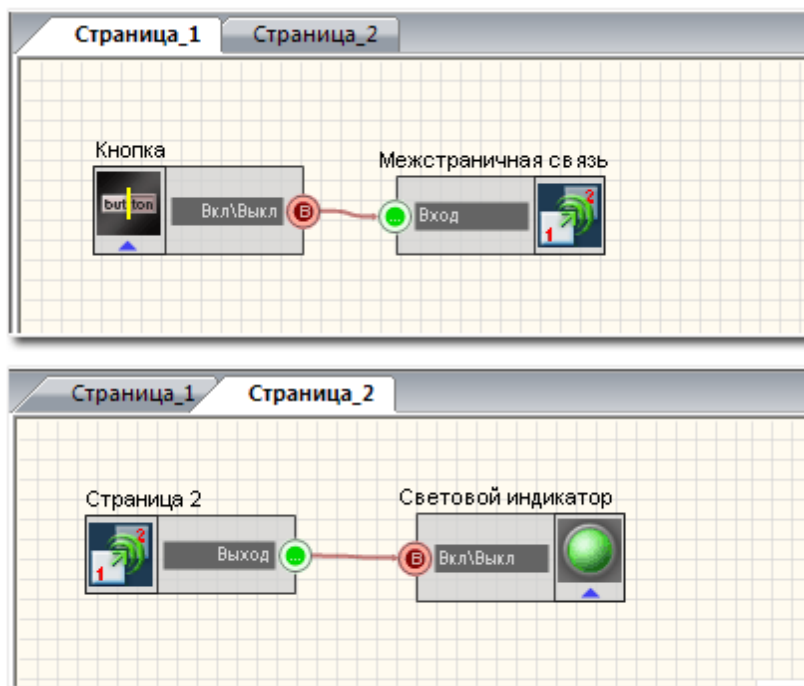


После, на передающем компоненте с помощью контекстного меню выбрать пункт "Редактировать компонент", далее появится меню с выбором принимающих компонентов "Межстраничная связь":

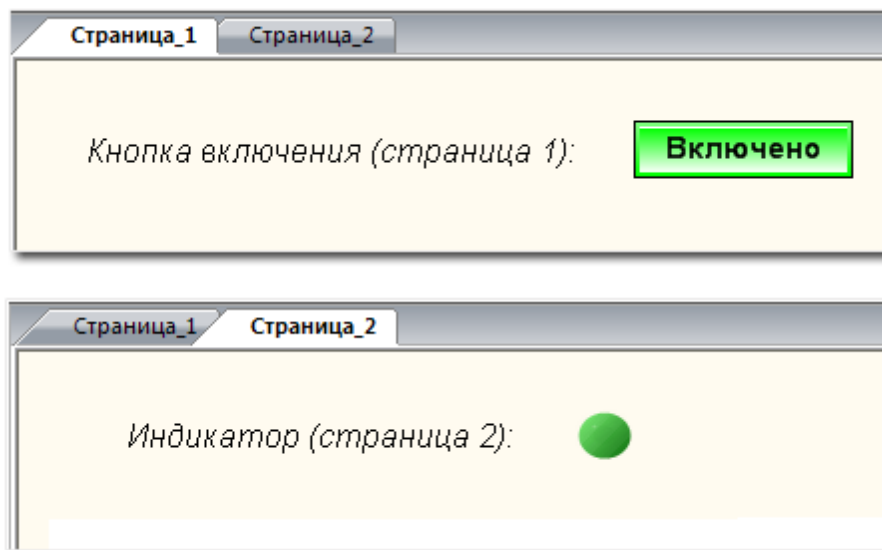


### *Пример*

### *Проект в SCADA ZETView*



### *Результат работы проекта*




Данный пример непосредственно в ZETView

## 23.7.Ключ

**Ключ** - компонент. Разрывает или поддерживает соединение, в зависимости от состояния на управляющем входе.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вход - входной контакт, на который могут передаваться данные любого типа.
- Вкл.\Выкл. - управляющий импульс. Если на данный вход подается постоянно "1" - ключ замкнут и данные со входа передаются на выход, если "0" - ключ разомкнут и на выход не поступает никаких данных.

#### ➤ Выходные

- Выход - выходной контакт, с которого передаются данные, полученные со входа, при замкнутом ключе.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

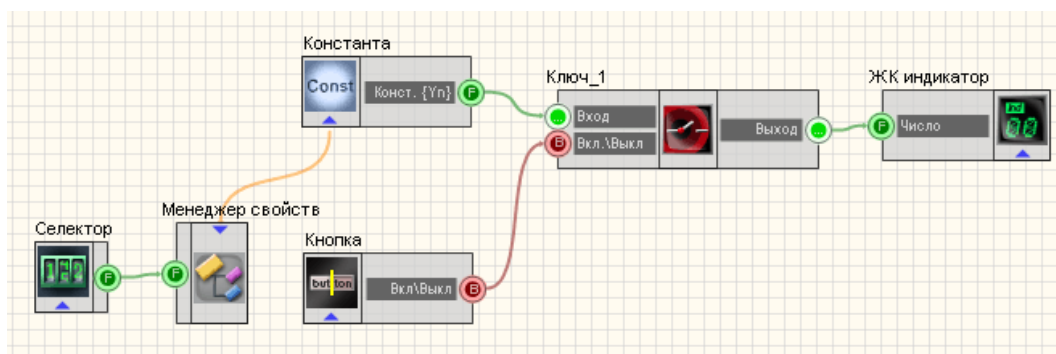
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Status (false) - активация/деактивация ключа текущее состояние.



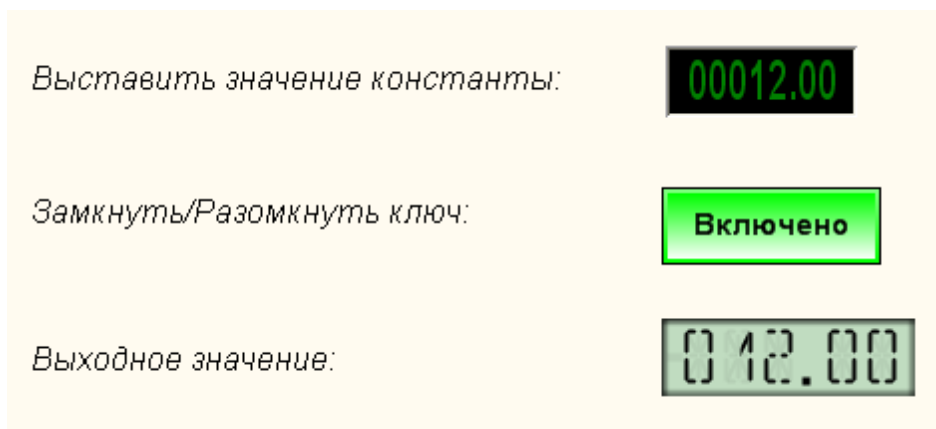
### *Пример*

### *Проект в SCADA ZETView*



На вход компонента "[Ключ](#)<sup>[533]</sup>" подается [Константа](#)<sup>[205]</sup>, компонент управляется [Кнопкой с фиксацией](#)<sup>[366]</sup>. Для того, чтобы получить доступ к дополнительным входам компонента [Константа](#)<sup>[205]</sup>, к нему подключен [Менеджер свойств](#)<sup>[534]</sup>, с помощью которого становится доступен вход, для настройки значения передаваемой константы с помощью [селектора](#)<sup>[213]</sup>.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## 23.8. Менеджер свойств

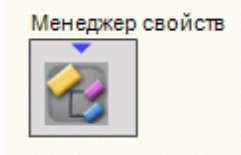
**Менеджер свойств** - компонент. Предназначен для управления свойствами других компонентов.

Каждый компонент SCADA системы ZETView имеет контактные площадки для взаимодействия с другими компонентами. Также, каждый компонент имеет различные свойства (цвет цифр индикатора, точность отображения численных значений



цифрового индикатора, время накопления данных измерительных приборов и т.д. и т. п.). Обычно, свойства компонента задаются на стадии проектирования и больше не изменяются. В ситуациях, когда необходимо управлять каким-либо свойством компонента, а соответствующей контактной площадки у компонента не предусмотрено, используется Менеджер свойств.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

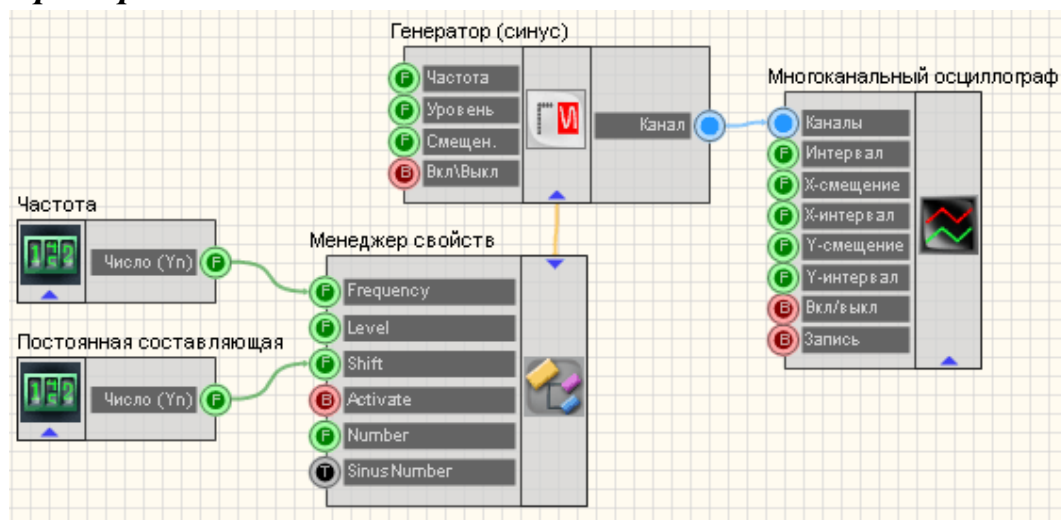
- Входные параметры полностью повторяют настраиваемые свойства компонента, к которому подключен Менеджер свойств.

##### ➤ Частные свойства:

Не имеет.



### Пример



### Проект в SCADA ZETView

В этой схеме компонент [Менеджер свойств](#)<sup>[534]</sup> открывает доступ для других модулей к частным свойствам [Синусоидальный сигнал](#)<sup>[247]</sup>. [Селекторы](#)<sup>[213]</sup> служат для задания частоты и постоянной составляющей синусоидального сигнала. [Многоканальный осциллограф](#)<sup>[494]</sup> служит для

графического отображения сигнала генератора.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 23.9.Менеджер стандартных свойств

**Менеджер стандартных свойств** - компонент. Предназначен для управления стандартными свойствами расположения компонента (Left, Top, Width, Height).

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

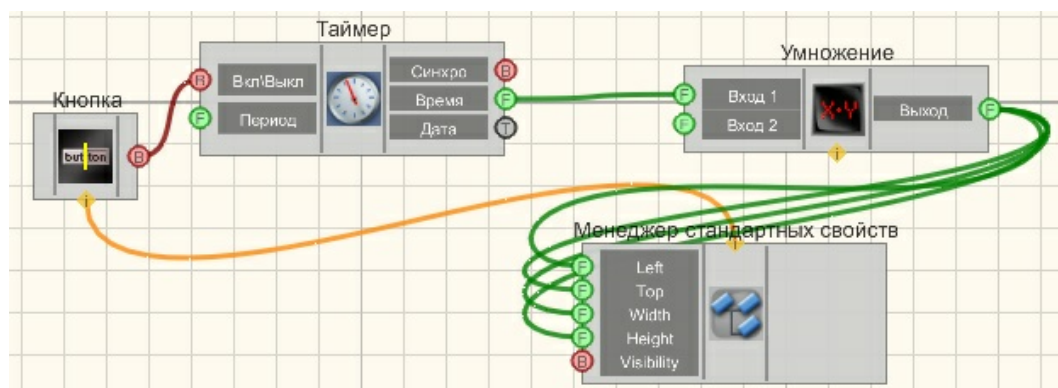
**Параметры:**

## ➤ Входные:

- Left - устанавливается координата X левого верхнего угла компонента (относительно левой стороны страницы)
- Top - устанавливается координата Y левого верхнего угла компонента (относительно левой стороны страницы)
- Width - устанавливается ширина компонента
- Neigth - устанавливается высота компонента
- Visibylity - устанавливается видимость компонента.

## ➤ Частные свойства:

Не имеет.

**Пример****Проект в SCADA ZETView**

В этой схеме компонент [Менеджер стандартных свойств](#)<sup>534</sup> позволяет управлять координатами компонента и общими свойствами. Компонент, изменяясь движется по экрану.

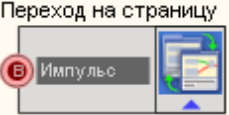
**Результат работы проекта**

Данный пример непосредственно в ZETView

## 23.10.Переход на страницу

**Переход на страницу** - компонент. При получении управляющего импульса на вход, переходит на указанную страницу.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - импульс перехода на другую страницу проекта.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства:

- pagenametogo - Имя страницы, на которую осуществляется переход.



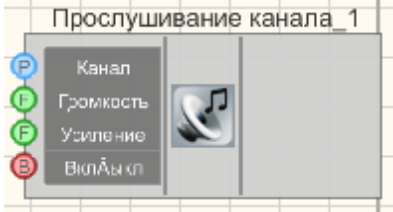
### *Пример*

Данный пример непосредственно в ZETView

## 23.11.Прослушивание канала

**Прослушивание канала** - компонент. Предназначен для воспроизведения звука, идущего с каналов.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входы:

- Канал - ID-сигнального канала.
- Громкость - уровень громкости.
- Усиление - усиление сигнала.
- Вкл\Выкл - входной контакт, отвечающий за включение и выключение прослушивания канала.
- 

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

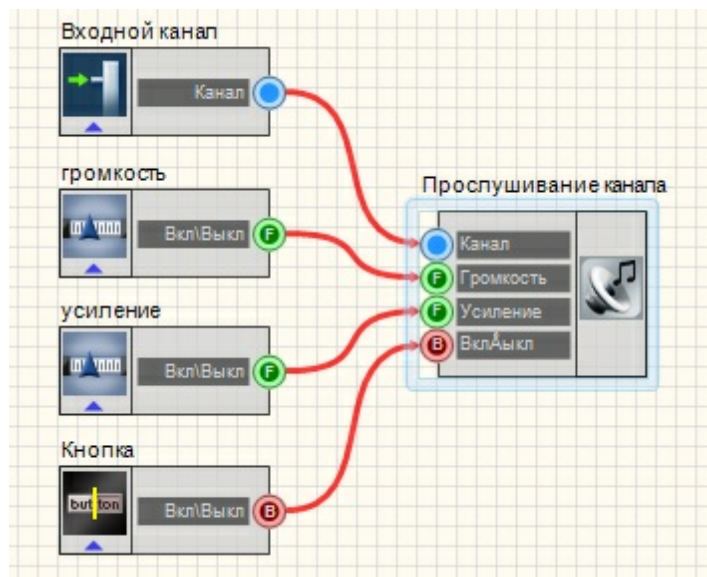
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Volume (50) - громкость.
- Gain (0) - усиление, дБ.
- Activate (false)- состояние прослушивания (включен или отключен).

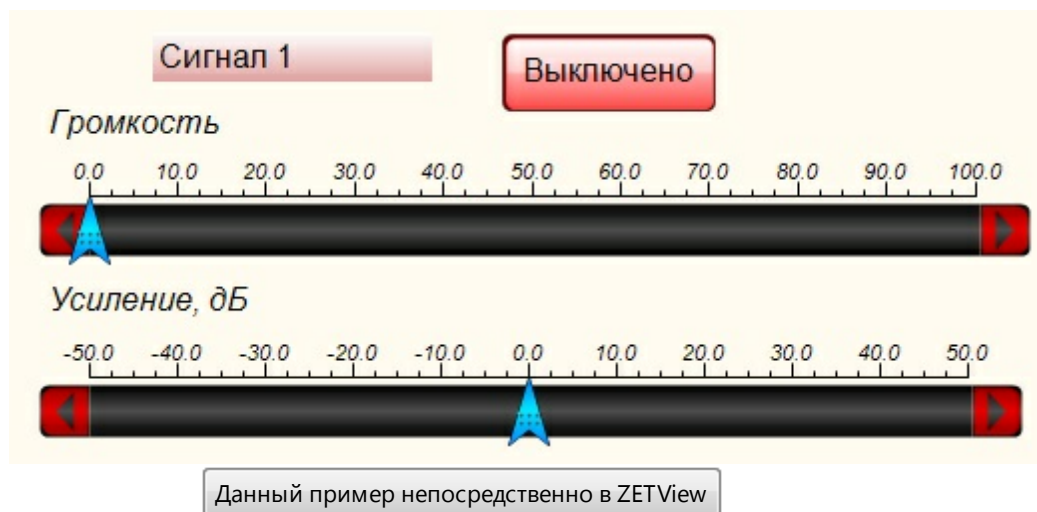


*Пример*

*Проект в SCADA ZETView*



### Результат работы проекта



## 23.12.Случайное число

**Случайное число** - компонент. Предназначен для создания случайных чисел.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вкл\Выкл - входной контакт, отвечающий за включение и выключение случайных чисел.

#### ➤ Выходные:

- Выход - на выходе получается случайное число.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Activate (true) - состояние работы генератора случайных чисел
- MinimumOfDistribution (-1) - минимальное значение распределения.
- MaximumOfDistribution (1) - максимальное значение распределения



### Пример

#### Проект в SCADA ZETView



### Результат работы проекта



Данный пример непосредственно в ZETView

## Глава 24. Регистратор

### 24.1. Автономный регистратор

<Раздел находится в разработке>

**Автономный регистратор** - компонент. Предназначен для создания сценария работы автономного регистратора: синхронизации, обмена данными и настройки режимов записи

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Вводные:

- Синхро - синхронизовать часы регистратора с часами компьютера.
- Запись - текст для записи в лог-файл.
- Записать - команда на запись в лог-файл.



- Читать - команда на чтение файла с носителя автономного регистратора.
  - Удалить - удалить файл с регистратора.
  - Список - получить список файлов с регистратора.
  - GPS - включить задержку на включение GPS, с
  - Длит. - установить длительность регистрации, с
  - Начало - установить время начала регистрации.
  - Получить - получить значение переменных.
- Выходные
- Данные - данные лог-файла.
  - Список - список файлов регистратора.
  - GPS - значение задержки на включение GPS, с.
  - Длит. - длительность регистрации, с.
  - Начало - время начала регистрации.
  - Время - внутреннее время регистратора.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- LogFileName (LOG.log) - имя лог-файла.
  - CompFileName - имя файла на компьютере
- 

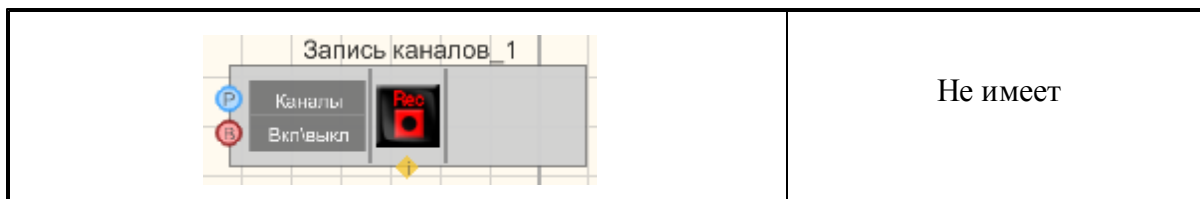
## 24.2.3 Запись каналов

<Раздел находится в разработке>

**Запись каналов** - компонент. Предназначен для непрерывной регистрации (записи) сигналов с каналов АЦП в файл.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Входные:

- Каналы - список записываемых каналов.
- Вкл/выкл - включение/выключение записи сигналов.

#### ➤ Выходные

Не имеет.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Duration (100) - устанавливается длительность записи, с.
- Preamble (0) - длительность преамбулы, с
- ChannelList - текущий список каналов, если каналов не выбрано, то по умолчанию всё.
- ExcludeList - если в канале встречается часть введённых через ; слов, то он игнорируется.
- Infinite (false) - непрерывная запись.
- UnitVisible (false) - видимость запускаемой программы writer.exe.
- ReactionTime (0) - время отложенного пуска, с.
- Activate (true) - состояние работы.

## Глава 25. Решения

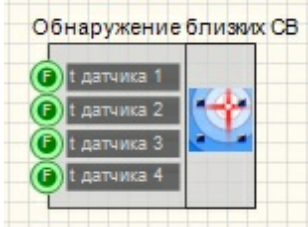
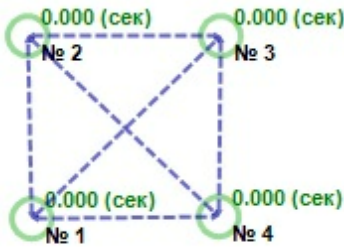
### 25.1. Обнаружение близких СВ

<Раздел находится в разработке>

**Обнаружение близких СВ** - компонент. Предназначен для локации места сейсмического воздействия по 4-м сейсмодатчикам.

Работает для области внутри четырёхугольника, образованного положением сейсмодатчиков, и на близком расстоянии от неё

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

**Параметры:**

## ➤ Входные:

- t датчика 1 - время сигнала от первого датчика, с.
- t датчика 2 - время сигнала от второго датчика, с.
- t датчика 3 - время сигнала от третьего датчика, с.
- t датчика 4 - время сигнала от четвертого датчика, с.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

Не имеет.

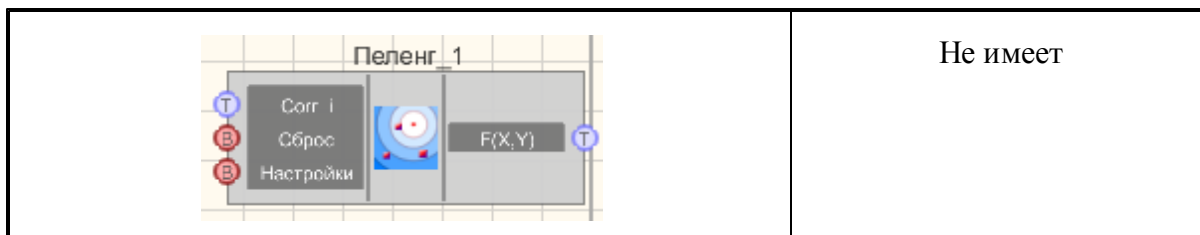
**25.2.Пеленг**

<Раздел находится в разработке>

**Пеленг** - компонент. Предназначен для определения местоположения источника сейсмических волн.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора

**Параметры:**

## ➤ Вводные:

- сорг\_1 - массив входных значений корреляционной функции.
- Сброс - обнулить массив выходной функции  $F(x,y)$
- Настройки - настройка координат расположения датчиков.

## ➤ Выходные

- $F(X,Y)$  - двумерный массив выходных значений.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

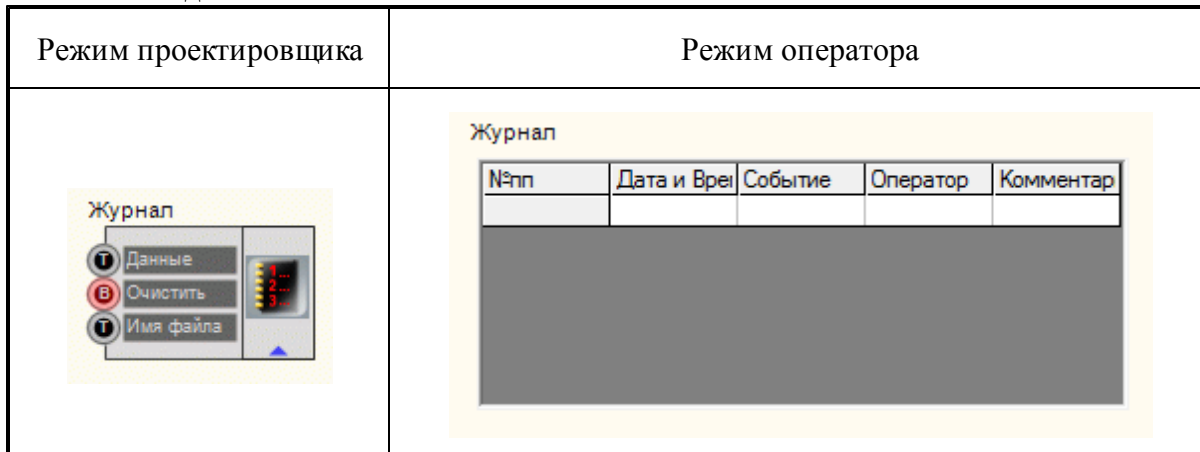
## ➤ Частные свойства:

Не имеет.

## Глава 26. Сообщения оператору

### 26.1. Журнал событий

**Журнал событий** - компонент. Служит для фиксации и представления событий в форме таблицы-журнала.

**Внешний вид компонента:**

**Параметры:**

## ➤ Входные:

- Данные - контакт, канал записи данных в журнал.
- Очистить - контакт, при подаче на который импульса, происходит очистка журнала.
- Имя файла - контакт, через который выставляется название файла сохранения.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- m\_sFileName (C:\TextLog.txt) - выбор необходимого файла, в который будет произведена запись событий.
- ColumnHeaders (#;Событие;Время;) - последовательное перечисление заголовков столбцов через знак-разделитель.

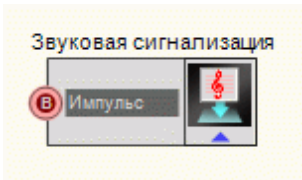
**Пример**

*Пример* реализации фиксации изменений: (находится в Примере компонента [Фиксация изменений](#)<sup>[194]</sup>)

**26.2.3 Звуковая сигнализация**

**Звуковая сигнализация** - компонент. Служит для реализации звукового сигнала, привлекающего внимание оператора.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Импульс - контакт, при подаче на который импульса, происходит проигрывание звукового файла.

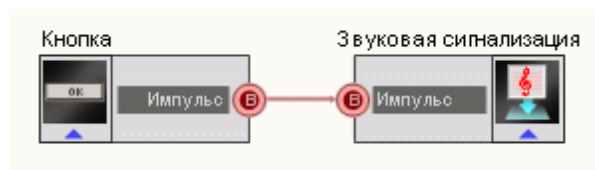
**Настраиваемые свойства:**

➤ Общие свойства (окружение):

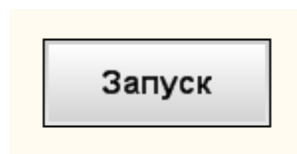
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства:

- FileName - выбор файла, который будет проигрываться по поступлению импульса.

**Пример****Проект в SCADA ZETView**

В этой схеме компонент Звуковая сигнализация производит проигрывание указанного звукового файла по нажатию [Кнопки без фиксации](#)<sup>[367]</sup>.

**Результат работы проекта**

Данный пример непосредственно в ZETView

**26.3.Маркер качества (к журналу событий)**

**Маркер качества (к журналу событий)** - компонент. Служит для вывода срочных сообщений оператору в виде специальной формы, возникающей на экране и привлекающей внимание.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входные:

- Строка - вывод, к которому подключается строка, которая необходим для маркера качества.
- Маркер - маркер качества

## ➤ Выходные

- Сообщение - объединение строки и маркера качества.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

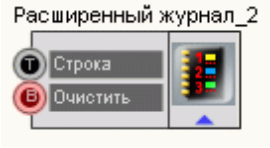

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- AddTime (true) - добавлять или не добавлять метку времени к строке.
- AddDate (true) - Добавить или не добавить метку даты к строке.
- NormalColor (ffffff) - цвет обычного текста для вывода.
- EventColor (ffed0f) - цвет текста в случае возникновения события, требующего внимания оператора.
- AlarmColor (ff0000) - цвет текста в случае возникновения ошибки или аварии, требующих внимание оператора.
- PerfectColor (00ff00) - цвет текста, в случае перехода из ошибки в нормальное состояние.
- divider (Не добавлять символ) - разделитель:
  1. Не добавлять символ.
  2. Табуляция.
  3. Пробел.

## 26.4.Расширенный журнал

**Расширенный журнал** - компонент. Служит для реализации звукового сигнала, привлекающего внимание оператора.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Строка - контакт, на который подается форматированная текстовая строка.
- Очистить - контакт, при подаче на который импульса, производится очистка журнала.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

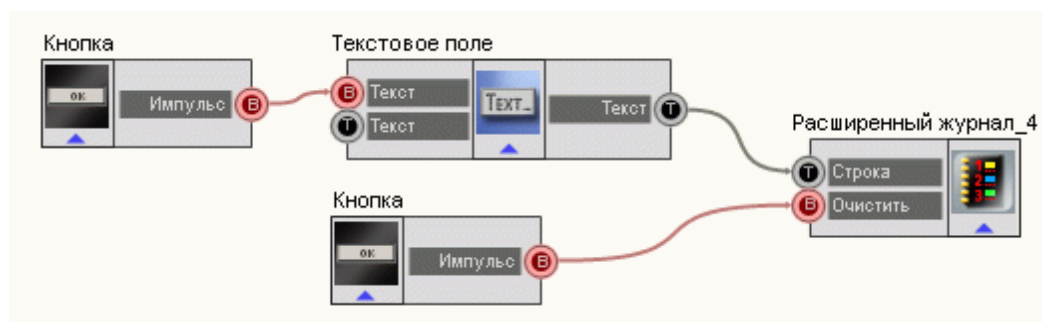
- Columns - выставление заголовка таблицы.
- Widths - установка ширины столбцов.
- Delimiter - установка знака-разделителя, при помощи которого имеется возможность форматировать таблицу (выставлять названия столбцов и их ширину).
- autoincrement (Нет) - автоинкремент первого столбца:
  1. Да - в этом режиме первый столбец заполняется номерами автоматически, увеличивая каждый последующий на единицу.
  2. Нет - в этом режиме автоинкремент отключен.
- SpecialKey - установка ключевого слова, при нахождении которого вся строка будет выделяться одним цветом.
- SpecialColor (ffff00) - выбор цвета, которым будет выделяться строка.



### *Пример*

### *Проект в SCADA ZETView*

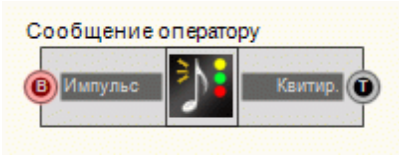
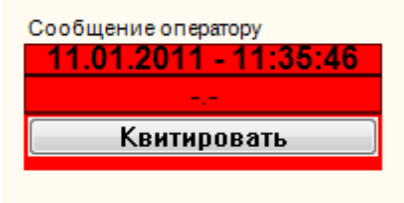




В этой схеме компонент Расширенный журнал позволяет формировать структурную запись входящих данных в виде таблицы. [Кнопка без фиксации](#)<sup>[367]</sup>, подключенная к [Текстовому полю](#)<sup>[561]</sup> подает синхроимпульс, по которому вся текстовая информация из компонента передается в Расширенный журнал. Оставшаяся Кнопка без фиксации позволяет очищать содержимое всей таблицы.

### *Результат работы проекта*



Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Входные:

- Импульс - на данный контакт подается импульс, вызывающий появления специальной формы сообщения для оператора.

#### ➤ Выходные

- Квитирование - выходной канал, с которого поступает текстовая информация при подтверждении приема сообщения оператором.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

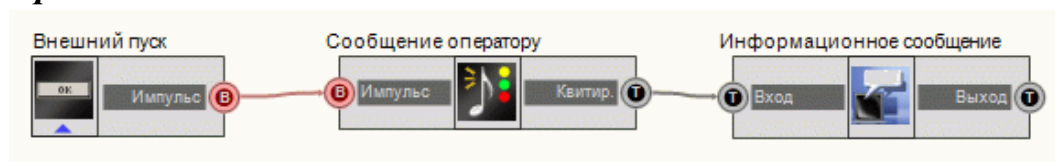
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- CautionTime (0) - время привлечения внимания сообщением в миллисекундах, мс.
- WindowColor (#0000) - установка цвета окна.
- String (-.-) - содержание строки сообщения.
- HeadHeight (12) - высота заголовочного текста.
- TextHeight (12) - высота основного текста.
- AllowSound (true) - включение/выключение звукового сопровождения сообщения.
- ButtonCaption (Квитировать) - установка названия кнопки



## Пример

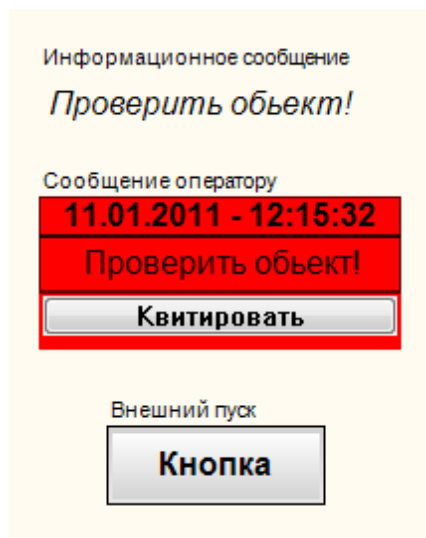
### Проект в SCADA ZETView



В этой схеме компонент [Сообщение оператору](#)<sup>[552]</sup> выводит на дисплей специальное информационное письмо оператору. [Кнопка без фиксации](#)<sup>[367]</sup> нужна для запуска данного спецсообщения. [Метка](#)<sup>[554]</sup> служит

для вывода более содержательного письма после квитирования [Сообщения оператору](#)<sup>552</sup>.

### *Результат работы проекта*



Данный пример непосредственно в ZETView

## Глава 27.Текст

### 27.1.Метка

**Метка** - компонент. Предназначен для ввода и вывода текста на дисплее.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
<p>Метка</p>	<p>Метка</p> <p><i>Текст метки</i></p>

#### Параметры:

##### ➤ Входные:

- Вход - контакт, на который поступает текстовая строка, надпись метки

##### ➤ Выходные:

- Выход - контакт, с которого поступает введенная в компонент текстовая строка,

надпись метки

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

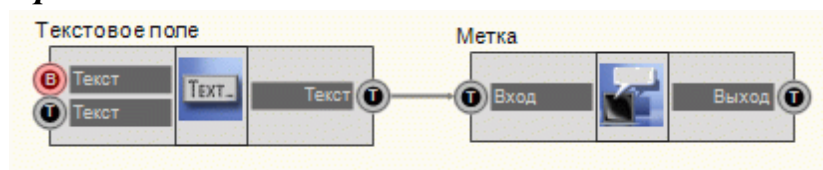
- TextSize (12) - размер текста.
- textstyle (FontStyleItalic) - стиль текста
- Transparent (true) - включение/выключение прозрачности.
- TextColor (ffffff) - цвет текста.
- Text (Текст метки) - надпись, выводимая на дисплей.
- alignment (По левому краю) - тип выравнивания текста:
  1. По левому краю.
  2. По середине.
  3. По правому краю.
- Frame (false) - включение/выключение рамки вокруг текста.



### Пример

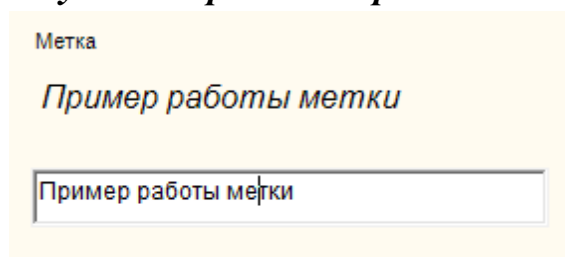
➤ Пример № 1 (Использование Метки в качестве выходного компонента).

### Проект в SCADA ZETView



В этой схеме компонент **Метка** служат для графического представления текстовой информации, которая поступает с [Текстового поля](#)<sup>[561]</sup>.

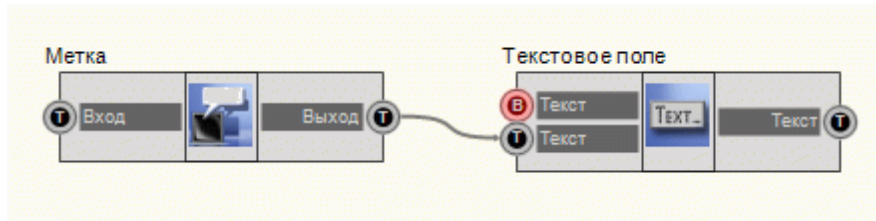
### Результат работы проекта



Данный пример непосредственно в ZETView

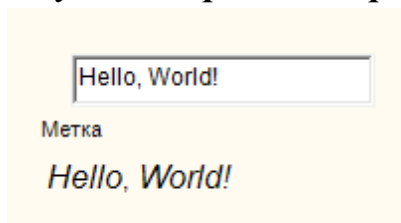
➤ Пример № 2 (Использование Метки в качестве входного компонента).

### Проект в SCADA ZETView



В этой схеме компонент Метка служат для передачи текстовой строки в Текстовое поле.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 27.2. Поиск слова в строке

**Поиск слова в строке** - компонент. Предназначен для поиска на соответствие набора символов в указанной текстовой строке.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

➤ Входные:

- Строка - вывод, к которому подключается строка, в которой необходимо провести поиск.
  - Слово - вывод, на который подается набор символов (слово), которое нужно найти в строке.
- Выходные:
- Символ - вывод, с которого поступает номер символа в случае совпадения.
  - Импульс - вывод, на котором устанавливается высокий или низкий логический уровень в зависимости от того найдено слово или нет.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

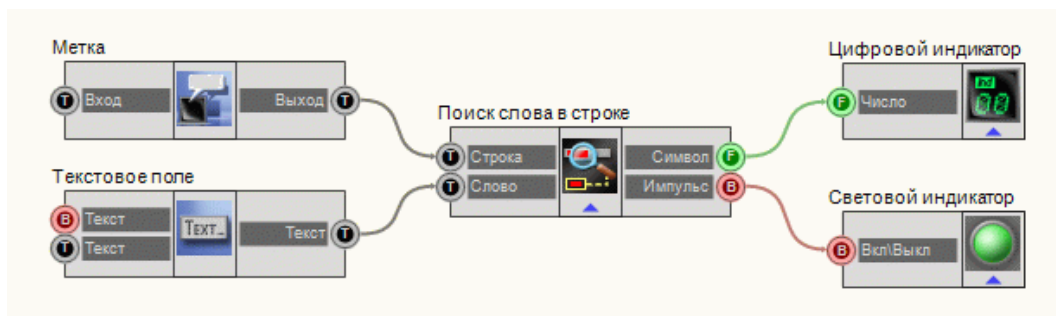
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- String - к данному выводу подключается строка, в которой будет происходить поиск.
- Word - слово, которое будет искаться в подключенной строке.
- FirstReaction (True) - установка реакции изменения выходного сигнала на изменение значений на входе №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (True) - установка реакции изменения выходного сигнала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- reactiontype (Оба события) - установка условия отправки единичного импульса с выхода:
  1. Соответствие - отправка импульса в случае нахождения введенного слова в строке.
  2. Несоответствие - отправка импульса в случае отсутствия введенного слова в строке.
  3. Оба события - отправка импульса в случае изменения в канале строки или в канале слова.



### *Пример*

### *Проект в SCADA ZETView*



В этой схеме компонент [Поиск слова в строке](#)<sup>[556]</sup> служит для поиска слова, которое вводится в [Текстовом поле](#)<sup>[561]</sup>. В случае обнаружения такого, [Световой индикатор](#)<sup>[356]</sup> становится зеленым, а на [Цифровом индикаторе](#)<sup>[362]</sup> высвечивается номер символа в строке.

### Результат работы проекта

*Метрология — наука об измерениях физических величин, методах и средствах обеспечения их единства и способах достижения требуемой точности*

Поиск слова:

Слово найдено:



Номер слова:

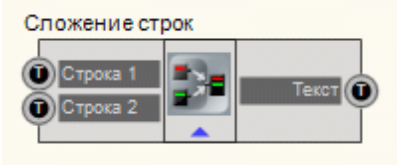
Данный пример непосредственно в ZETView

## 27.3.Сложение строк

**Сложение строк** - компонент. Предназначен для соединения двух независимых строк в одну.

**Внешний вид компонента:**



Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Строка 1 - вывод, на который поступает текстовая строка №1.
- Строка 2 - вывод, на который поступает текстовая строка №2.

#### ➤ Выходные:

- Текст - вывод, с которого выводится результирующая текстовая строка (уже склеенная).

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Text - текущий результат сложения.
- FirstString - первая строка-слагаемое.
- SecondString - вторая строка слагаемая.
- FirstReaction (True) - установка реакции изменения выходного канала на изменение значений на входе Строки №1:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- SecondReaction (True) - установка реакции изменения выходного канала на изменение значений на входе №2:
  1. True - при изменении сигнала на входе - изменится выходное значение.
  2. False - при изменении сигнала на входе - не изменится выходное значение.
- specialdivider (Табуляция) - добавление между строк одного специальных символов-разделителей:
  1. Табуляция - добавляет горизонтальную табуляцию между строками.
  2. Пробел - добавляет одинарный пустой интервал между строками.
  3. Перенос строки - добавляет вторую строку к первой, но начинает ее с новой строки.
- SpecSymbol (False) - включение и отключение спецсимволов:
  1. True - в этом режиме между строками добавляется спецсимвол, установленный в свойстве specialdivider.
  2. False - в этом режиме между строк добавляется символ, указанный в

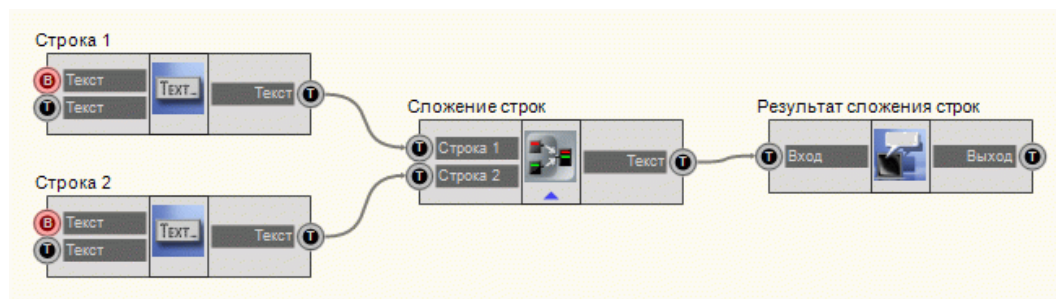
свойстве Divider.

- Divider - символ-разделитель, устанавливаемый пользователем.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент Сложение строк служит для преобразования двух независимых строк в единую. [Метка](#)<sup>[554]</sup> нужна для вывода результирующей строки. [Текстовые поля](#)<sup>[561]</sup> нужны для ввода двух независимых строк, которые подвергнутся операции конкатенации (склеивания) строк.

### Результат работы проекта



Данный пример непосредственно в ZETView

## 27.4.Статический текст Windows стиль

**Статический текст Windows стиль** - компонент. Предназначен для вывода статического текста в режиме оператора

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
Не имеет	<div style="border: 1px solid black; padding: 5px; background-color: #ffffcc;"> Текст  <i>Комментарий</i> </div>

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства:

Не имеет.

**Пример**

Пример создания переключателя выходов: (находится в Примере компонента [Комбинированный список](#)<sup>203</sup>)

**27.5.Текстовое поле**

**Текстовое поле** - компонент. Предназначен для ввода и вывода текстовой информации на дисплее.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
<div style="border: 1px solid black; padding: 5px; background-color: #ffffcc;"> Текстовое поле  </div>	<div style="border: 1px solid black; padding: 5px; background-color: #ffffcc;"> </div>

**Параметры:**

➤ Входные:

- Текст - контакт, при подаче на который синхроимпульса, идет передача текста на выход.
- Текст - контакт, к которому подключается текстовая строка, которая будет выведена компонентом.

➤ Выходные:

- Текст - контакт, с которого поступает введенная в компонент текстовая строка.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Text - текст внутри поля.
- type (По изменению) - тип вывода текста:
  1. По изменению - вывод текста на дисплей в случае изменения состояния на входе.
  2. По импульсу - вывод текста на дисплей в случае подачи на вход импульса.

**Пример**

*Пример* создания ввода и передачи текстовой строки: (находится в Примере компонента [Метка](#)<sup>[554]</sup>)

**27.6.Разделитель строки**

**Разделитель строки** - компонент. Разбивает входную строку на несколько строк, сигналом для разделения строки служит указанный символ.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Вход - входная строка.

➤ Выходные:

- Выход 1 - Выход 49 - выходные строки. Количество зависит от параметра Outputs. По умолчанию выходов два.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

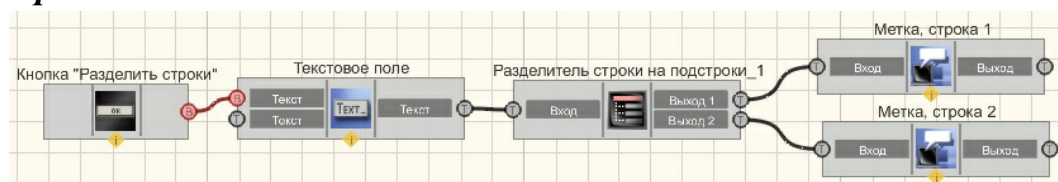
➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Delimiter (|) - символ разделения строки. Не отображается в разделенных строках. По умолчанию символ "|"
- Outputs (2) - количество выходов. По умолчанию 2. Максимальное количество 49.



## Пример

### Проект в SCADA ZETView



В данном примере рассмотрена работа компонента "[Разделитель строки](#)"<sup>[562]</sup>.

Пользователем вводится строка в [Текстовое поле](#)<sup>[561]</sup>. После получения управляющего импульса с [кнопки без фиксации](#)<sup>[367]</sup> под названием "Разделить строки", введенная строка передается на компонент "[Разделитель строки](#)"<sup>[562]</sup>, где строки разделяются при получении символа "|". Результат выводится на компонент "[Метка](#)"<sup>[554]</sup>.

### Результат работы проекта

*Разделитель строки - символ |*

*Ввод строки:*

ZETView компонент "Разделитель строки" | Тест

Разделить строки

*Результат:*

*Строка 1:*

ZETView компонент "Разделитель строки"

*Строка 2:*

Тест

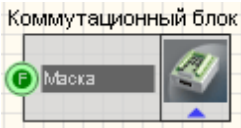
Данный пример непосредственно в ZETView

## Глава 28. Устройства ZET

### 28.1. Коммутационный блок

**Коммутационный блок** - компонент. Предназначен для управления состоянием реле коммутационного блока, подключенного к цифровому выходу модулей АЦП/ЦАП. Предназначен для управления состоянием реле, находящихся на коммутационной колодке, при помощи цифрового порта модулей АПЦ-ЦАП ZET 210, ZET 220, ZET 230.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Маска - значение маски состояния реле.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Combination (0) - состояние всех реле.
- DeviceNum () - порядковый номер устройства.

## 28.2.ZET 0xxx2

**ZET 0xxx2** - компонент. Предназначен для измерения параметров спектральных составляющих сигналов с различных датчиков, корреляционной структуры сигналов, генерации электрических сигналов с нормированными метрологическими параметрами, измерения уровней шума и уровней вибрации в 1/3-октавных полосах, что позволяет использовать его в качестве виброметра и шумомера. Используется для визуализации, сбора данных и другого вида обработки сигналов с различных датчиков: [вибропреобразователей](#), [измерительных микрофонов](#), [гидрофонов](#) со встроенными усилителями ИСР и т.п. Компонент необходим для контрольно-измерительных и диагностического оборудования при проведении различного рода измерений и испытаний, для мониторинга длительных процессов, носящих как стационарный, так и динамический характер.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входные:

- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.

- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.
- Выходные
  - Номер - номер устройства.
  - КУ - коэффициент усиления текущего канала.
  - Ч.АЦП - частота дискретизации АЦП.
  - Ч.ЦАП - частота дискретизации ЦАП.
  - Канал - выбранный канал анализатора.

#### Настраиваемые свойства:

- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>92</sup>
- Частные свойства:
  - Не имеет.

## 28.3.ZET 0xxx4/0xxx8

**ZET 0xxx4/0xxx8** - компонент. Предназначен для измерения параметров спектральных составляющих сигналов с различных датчиков, корреляционной структуры сигналов, генерации электрических сигналов с нормированными метрологическими параметрами, измерения уровней шума и уровней вибрации в 1/3-октавных полосах, что позволяет использовать его в качестве виброметра и шумомера. Используется для визуализации, сбора данных и другого вида обработки сигналов с различных датчиков: [вибропреобразователей](#), [измерительных микрофонов](#), [гидрофонов](#) со встроенными усилителями ИСР и т.п. Компонент необходим для контрольно-измерительных и диагностического оборудования при проведении различного рода измерений и испытаний, для мониторинга длительных процессов, носящих как стационарный, так и динамический характер.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:



➤ Входные:


- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.
- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.

➤ Выходные

- Номер - номер устройства.
- КУ - коэффициент усиления текущего канала.
- Ч.АЦП - частота дискретизации АЦП.
- Ч.ЦАП - частота дискретизации ЦАП.
- Канал - выбранный канал анализатора.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) 

➤ Частные свойства:

Не имеет.

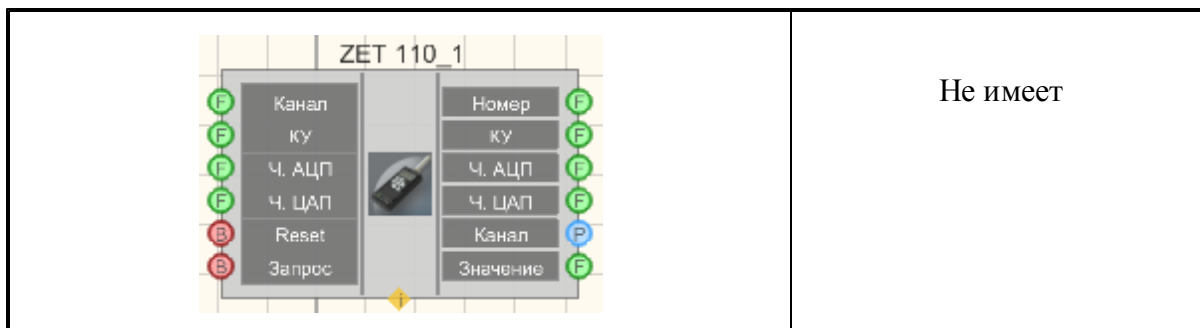
---

## 28.4.ZET 110

**ZET 110** - компонент. Предназначен для измерения параметров сигналов с различных датчиков, измерения уровней шума и уровней вибрации, отображения измеряемых параметров на экране и непрерывной записи сигналов во встроенную энергонезависимую память объемом до 2 Гб.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
----------------------	-----------------



Не имеет

### Параметры:

#### ➤ Входные:

- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.
- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.
- Запрос - запрос измеренного значения.

#### ➤ Выходные

- Номер - номер устройства.
- КУ - коэффициент усиления текущего канала.
- Ч.АЦП - частота дискретизации АЦП.
- Ч.ЦАП - частота дискретизации ЦАП.
- Канал - выбранный канал анализатора.
- Значение - канал, выдающий измеренное значение.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>


#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- turcalc (PEAK)- тип вычисляемых величин.
- fltrnm (HP1)- установка фильтра.
- device (Шумомер) - тип устройства:
  1. Шумомер.
  2. Виброметр.
  3. Регистратор.

## 28.5.ZET 210

**ZET 210** - компонент. Предназначен для измерений параметров сигналов в широком частотном диапазоне (с частотой дискретизации до 400 кГц), поступающих с различных первичных преобразователей. Цифровой (разъем DB-15) и аналоговый выходы (разъем DB-25) могут использоваться в цепях управления различными исполнительными механизмами.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.
- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.

#### ➤ Выходные

- Номер - номер устройства.
- КУ - коэффициент усиления текущего канала.
- Ч.АЦП - частота дискретизации АЦП.
- Ч.ЦАП - частота дискретизации ЦАП.
- Канал - выбранный канал анализатора.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства:

Не имеет.

## 28.6.ZET 220

**ZET 220** - компонент. Предназначен для измерений параметров сигналов с высокой точностью и большим динамическим диапазоном, поступающих с различных первичных преобразователей: термосопротивлений, термопар, датчиков с универсальным токовым выходом 4...20 мА, акселерометров ВС201/202.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.
- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.

#### ➤ Выходные

- Номер - номер устройства.
- КУ - коэффициент усиления текущего канала.
- Ч.АЦП - частота дискретизации АЦП.
- Ч.ЦАП - частота дискретизации ЦАП.
- Канал - выбранный канал анализатора.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

#### ➤ Частные свойства:

Не имеет.

## 28.7.ZET 230

**ZET 230** - компонент. Предназначен для измерений параметров сигналов с высокой точностью, большим динамическим диапазоном и большим частотным диапазоном, поступающих с различных первичных преобразователей: акселерометров BC 201, BC 202, микрофонов BC 501.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал - установка текущего канала.
- КУ - установка усиления канала.
- Ч.АЦП - установка частоты дискретизации АЦП.
- Ч.ЦАП - установка частоты дискретизации ЦАП.
- Reset - Обновление параметров устройств.

#### ➤ Выходные

- Номер - номер устройства.
- КУ - коэффициент усиления текущего канала.
- Ч.АЦП - частота дискретизации АЦП.
- Ч.ЦАП - частота дискретизации ЦАП.
- Канал - выбранный канал анализатора.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

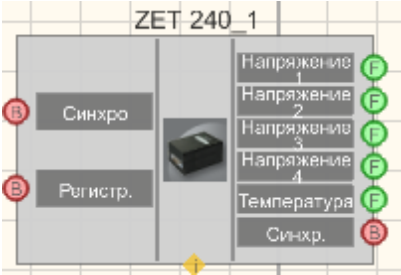
#### ➤ Частные свойства:

Не имеет.

## 28.8.ZET 240

**ZET 240 - компонент.** Предназначен для измерений и регистрации сигналов с сейсмоприемников и является средством измерения параметров электрического сигнала.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	<p>Не имеет</p>

### Параметры:

#### ➤ Входные:

- Синхро - канал для получения данных с устройства.
- Регистр. - канал для перехода в режим регистрации

#### ➤ Выходные

- Напряжение 1 - уровень напряжения на первом канале устройства, В.
- Напряжение 2 - уровень напряжения на втором канале устройства, В.
- Напряжение 3 - уровень напряжения на третьем канале устройства, В.
- Напряжение 4 - уровень напряжения на четвертом канале устройства, В.
- Температура - температура на устройстве.
- Синхр. - состояние дискретизации в GPS.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>


#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Device (0) - порядковый номер выбранного устройства.

## 28.9. Синхронный генератор ZET7090

**Синхронный генератор ZET7090** - компонент. ZET 7090 Generator-485 - аналоговый двухканальный синхронный генератор с интерфейсом RS-485. Предназначен для формирования сигналов синусоидальной, импульсной, пилообразной формы. 2 синхронных канала. Может применяться самостоятельно или в системах управления и автоматизации.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Частота А - канал частоты сигнала А генератора.
- Уровень А - канал уровня сигнала А генератора.
- Смещен. А - канал смещения сигнала А генератора относительно нуля.
- Вкл/выкл А - состояние работы А генератора.
- Частота В - канал частоты сигнала В генератора.
- Уровень В - канал уровня сигнала В генератора.
- Смещен. В - канал смещения сигнала В генератора относительно нуля.
- Вкл/выкл В - состояние работы В генератора.
- Фаза - фаза сигнала В генератора относительно сигнала А генератора.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):
  - Frequency\_A (1000) - частота сигнала А генератора, Гц.
  - Level\_A (1) - уровень сигнала А генератора, В
  - Shift\_A (0) - смещения сигнала А генератора относительно нуля, В.
  - Status\_A (false) - Включено (true)/Выключено (false) состояние работы А генератора.
  - Frequency\_B (1000) - частота сигнала В генератора, Гц.
  - Level\_B (1) - уровень сигнала В генератора, В
  - Shift\_B (0) - смещения сигнала второго генератора относительно нуля, В.
  - Status\_B (false) - Включено (true)/Выключено (false) состояние работы В генератора.
  - Phase (0) - фаза сигнала В генератора относительно сигнала А генератора, град.
  - serialNumH (0) - устанавливается старшая часть серийного номера
  - serialNumL (0) - устанавливается младшая часть серийного номера

## 28.10. Смена адреса ZET7xxx

**Смена адреса ZET7xxx** - компонент. Предназначен для смены адреса Zet7xxx датчиков.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Тек. адрес - устанавливается текущий адрес устройства.
- Сер. номер - устанавливается серийный номер преобразователя интерфейса.
- Тр. адрес - устанавливается требуемый адрес устройства.

#### ➤ Выходные

- Тек. состояние - текущее состояние смены адреса.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>



➤ Частные свойства (в скобках значение, установленное по умолчанию):

- CurrNode (0) - устанавливается текущий номер ноды.
- SerialNum (0) - устанавливается серийный номер преобразователя интерфейса
- NeedNode (0) - устанавливается требуемый адрес ноды.
- Channel1Name (Канал\_1) - устанавливается имя первого канала.
- Channel2Name (Канал\_2) - устанавливается имя второго канала.
- Channel3Name (Канал\_3) - устанавливается имя третьего канала.
- Channel4Name (Канал\_4) - устанавливается имя четвертого канала.

## 28.11.Событийный канал ZETxxx

**Событийный канал ZET7xxx** - компонент. Предназначен для чтения данных из событийных каналов модулей Zet7xxx.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Сер. номер. - устанавливается серийный номер интерфейсной платы.
- Нода - устанавливается номер ноды подключаемого устройства.
- Кол-во каналов- устанавливается количество событийных каналов.

➤ Выходные

- нет

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Ser\_number (0) - устанавливается серийный номер интерфейсной платы.
- Num\_node (0) - устанавливается номер ноды.
- Quan (0) - устанавливается количество каналов.

- WinSignal (False) - выборка сигнала с высоким разрешением (окно сигнала).
- NumPrevEvents (0) - устанавливается количество отображаемых последних событий. С его помощью можно настраивать отображения количества событий после запуска проекта.

## 28.12.Тензодатчик ZET7010

**Тензодатчик ZET7010 - компонент.** ZET 7010 Tensometer-485 интеллектуальный тензодатчик с интерфейсом RS-485 (статические измерения). Измерительный модуль ZET 7010 Tensometer-485 совместно с первичным преобразователем образует интеллектуальный датчик для измерения тензовеличин в статике. Это цифровой датчик, передающий результаты измерений по интерфейсу RS-485, используя протокол Modbus.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Калибровочная поправка. - устанавливается калибровочная поправка.
- Калибровочное смещение - устанавливается калибровочное смещение.

#### ➤ Выходные

- Тензосопротивление - текущее значение измеряемого тензосопротивления;
- Название канала - текущее название измеряемого канала подключенного датчика;
- Серийный номер устройства - текущий серийный номер подключенного устройства;
- Дата программного обеспечения - текущая дата программного обеспечения;
- Дата аппаратного обеспечения - текущая дата аппаратного обеспечения;

- Адрес устройства - текущий адрес устройства.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- `schemetype` (Мост) - устанавливается схема подключения тензорезистора (мост/полумост);
- `freq` (5) - устанавливается частота, Гц;
- `MeasureRange` (16) - устанавливается диапазон измерений (в ед. изм.);
- `Gain` (0) - устанавливается коэффициент передачи мВ/В;
- `StartValue` (0) - устанавливается начальное значение (в ед. изм.);
- `calcmethod` (Коэффициент передачи) - устанавливается метод расчёта (коэффициент передачи/калибровочная кривая);
- `Accuracy` (0,1) - устанавливается точность, %;
- `measure` (мВ/В) - устанавливается единица измерений (мВ/В; Н; кН; кг; т; Н/м);
- `SerialNum` (0) - устанавливается серийный номер датчика;
- `Calibr` (1) - устанавливается калибровочная поправка;
- `Offset` (0) - устанавливается калибровочное смещение;
- `DeviceSerialNum` (0x0000000000000000) - устанавливается серийный номер устройства.

Примечание;

Если задан нулевой серийный номер, то используется первый найденный датчик данного типа (т.е. ZET7010).

## 28.13. Термометр термопары ZET7020

**Термометр термопары ZET7020** - компонент. Предназначен для измерения температуры жидких, твердых, газообразных и сыпучих сред, неагрессивных к материалу корпуса термопары. Области применения: теплоэнергетика, химическая промышленность, металлургия. Используются в составе интеллектуальных датчиков температуры с интерфейсом RS-485.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



Не имеет

### Параметры:

#### ➤ Входные:

- Тип термопары - тип подключаемой термопары.
- Частота - частота выходного сигнала, Гц.
- Калибровочная константа АЦП - устанавливается калибровочная константа в устройство АЦП.
- Смещение АЦП - устанавливается калибровочное смещение в устройство АЦП.

#### ➤ Выходные

- Температура - текущая температура измерительного канала.
- Название - текущее название измерительного канала датчика ZET7020
- Серийный номер - текущий серийный номер датчика ZET7020
- Дата программы - текущая дата создания программного обеспечения датчика ZET7020
- Дата железа - текущая дата создания аппаратного обеспечения ZET7020
- Нода - текущий modbus адрес устройства ZET7020

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Туре (Напряжение) - тип подключаемого датчика к термометру термопары.
  1. Напряжение.
  2. Резистор.
  3. КХС
  4. Тип R
  5. Тип J
  6. Тип T
  7. Тип E
  8. Тип K
  9. Тип N
  10. Тип A1
  11. Тип A2
  12. Тип A3
  13. Тип L

## 14. Тип М

- freq (1) - устанавливается частота, Гц
- calibrADC (1) - калибровочная поправка коэффициента передачи АЦП
- offsetADC (0) - смещение напряжения, В
- channelVal (0) - текущее значение датчика подключенного к термометру термопары, ед. изм.

## 28.14. Термометр термосопротивления ZET7021

**Термометр термосопротивления ZET7021** - компонент. Предназначен для измерения температуры. Предназначены для измерения и контроля температуры жидких, твердых, газообразных и сыпучих сред в различных отраслях промышленности. Используются в составе интеллектуальных датчиков температуры с интерфейсом RS-485.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Козф. пер. - канал для калибровочной поправки коэффициента передачи.
- Смещение - канал смещения напряжения.
- Запись - запись калибровочных констант в устройство.

#### ➤ Выходные

- Канал- текущий измерительный канал.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

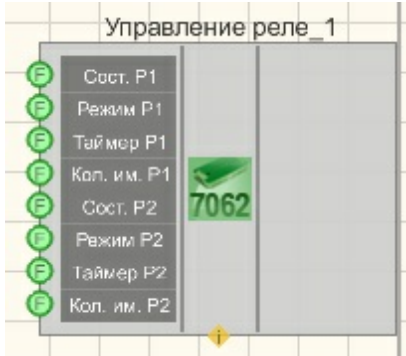
- Number (0) - порядковый номер выбранного устройства начиная с 0

## 28.15.Управление реле ZET7062

**Управление реле ZET7062** - компонент. Предназначен для управления реле. Настройки по умолчанию.

Сохраняется во флэш. Выставляет начальное состояние реле.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	<p>Не имеет</p>

### Параметры:

#### ➤ Входные:

- Сост. P1 - состояние реле - один раз изменяет состояние при режиме запуска.
- Режим P1 - режим запуска - единичное изменение состояния
- Таймер P1 - изменение состояния - по времени.
- Кол. им. P1 - количество импульсов - сколько раз реле изменит своё состояние при режиме таймер. (-1 - бесконечно)
- Сост. P2 - состояние реле - один раз изменяет состояние при режиме запуска.
- Режим P2 - режим запуска - единичное изменение состояния
- Таймер P2 - изменение состояния - по времени.
- Кол. им. P2 - количество импульсов - сколько раз реле изменит своё состояние при режиме таймер. (-1 - бесконечно)

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- stater1 (A) - состояние реле 1 (A/B)
- modestart1 (Старт) - режим запуска реле 1 (Старт/Таймер)
- timerR1 (0) - Таймер запуска реле 1, сек

- pulseNumR1 (0) - Количество импульсов реле 1, (-1) - постоянно
- stater2 (A) - состояние реле 2 (A/B)
- modestartR2 (Старт) - режим запуска реле 2 (Старт/Таймер)
- timerR2 (0) - Таймер запуска реле 2, сек
- pulseNumR2 (0) - Количество импульсов реле 2, (-1) - постоянно
- serialNumH (0) - устанавливается старшая часть серийного номера
- serialNumL (0) - устанавливается младшая часть серийного номера

## 28.16.Устройство ZET

**Устройство ZET** - компонент. Предназначен для подключения и настройки ZET приборов

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Канал АЦП - установка текущего канала АЦП.
- КУ - установка усиления канала.
- Ч.д.АЦП - установка частоты дискретизации АЦП.
- ICP - установка питания по ICP по каналу.
- Канал ЦАП - установка текущего канала ЦАП
- Аттен. - установка коэффициента ослабления по текущему каналу ЦАП.
- Ч.д.ЦАП - установка частоты дискретизации ЦАП
- Загрузка - Устанавливается обновление параметров устройств.

#### ➤ Выходные

- Название - Название устройства.
- Каналы - количество каналов АЦП устройства.
- КУ - устанавливается коэффициент по текущему каналу.

- Ч.д.АЦП - устанавливается текущая частота дискретизации АЦП
- Ч.д.ЦАП - устанавливается текущая частота дискретизации ЦАП.
- Канал - выбранный канал.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Device (0) - устанавливается порядковый номер устройства.

## 28.17. Цифровой генератор ZET7060g

**Цифровой генератор ZET7060g** - компонент. Предназначен для генерации импульсных сигналов с частотой до 100 МГц.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

➤ Входные:

- Частота - частота сигнала генератора.
- Скважность - устанавливается скважность импульсного сигнала.
- Вкл\Выкл - параметр, отвечающий за включение и выключение цифрового генератора.

#### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Frequency (240) - Частота генерируемых импульсов, Гц.
- Pososity (0.000960) - Скважность генерируемых импульсов;
- Enable (False) - Включение/Выключение цифрового генератора;
- SerialNumber (0x0000000000000000) - Серийный номер подключаемого

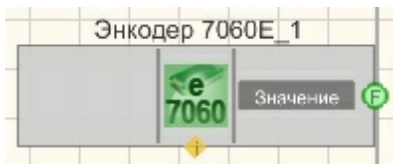
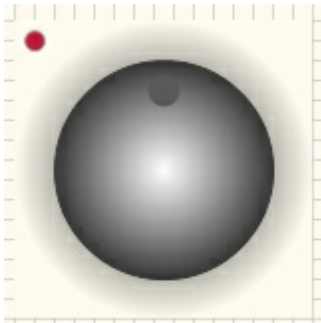


устройства, если к компьютеру будет подключен только один модуль ZET7060G, то серийный номер можно не указывать.

## 28.18.Энкодер ZET7060E

**Энкодер ZET7060E** - компонент. Предназначен для измерения угловых и линейных перемещений, а также определения скорости, ускорения и положения вращающейся оси/вала. Передача данных осуществляется по интерфейсу RS-485, используя протокол Modbus.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	

### Параметры:

#### ➤ Выходные:

- Значение - вывод, на котором выставляется управляющий сигнал, которое задаст пользователь.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- GradientBackColor (373737) - устанавливается цвет селектора.
- selectorform (Сфера) - определяется форма селектора:
  1. Цилиндр.
  2. Сфера.
- cursorform (Окружность) - определяется форма курсора:
  1. Треугольник.
  2. Окружность.

- CursorColor (005eff)- определяется цвет курсора (для стиля "С заливкой").
- cursorstyle (Градиент) - устанавливается стиль курсора:
  1. С заливкой
  2. Градиент
- shade (Темная) - устанавливается тень:
  1. Нет.
  2. Темная.
  3. Светлая.
- LinearVisible (true) - устанавливается видимость линейки (True/False).
- Linearstyle (Разной длины) - определяется стиль линейки:
  1. Одной длины
  2. Разной длины
- LinearColor (a0a0a4) - устанавливается цвет линейки.
- ValueOnLap (1) - определяется изменение значения при повороте на 360 градусов.
- CurrentValue (0) - устанавливается текущее значение.
- StepMove (true) - определяется движение по шагам (Вкл./Выкл.).
- ValueInStep (0.05)- устанавливается дискретность шага (не может быть больше, чем ValueOnLap / 8).
- showlist (Да) - Показывать выпадающий список с каналами во время работы программы (по нажатию на правую кнопку мыши)
- Channel (вх.канал) - определяется имя канала энкодера.

## Глава 29.Файл

### 29.1.Базы данных

**Базы данных** - компонент. Позволяет работать с базами данных SQL

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

## ➤ Входящие:

- Вход - служит для добавления строки в Бд.
- Поиск - позволяет осуществлять поиск записей в БД.
- Таблица - устанавливает имя в таблице.
- Вкл/Выкл - подключиться к Бд/отключиться от Бд.

## ➤ Выходные

- Состояние - состояние подключения.
- Данные - новая порция данных.
- Выход - значение столбца.
- Время - метки времени.
- Текст - текст выделенной строки.
- Номер - номер выделенной строки.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IP - IP адрес компьютера, на котором установлена база данных.
- DSN - имя источника данных DSN.
- Database - название базы данных, к которой необходимо произвести подключение.
- User - пользователь (имя, введенное при установке MySQL).
- Password - пароль пользователя.
- Table - таблица, к которой производим подключение.
- Delimiter (|) - текстовый разделитель столбцов (полей) базы данных.
- Column - столбец, значения которого будут на выходе.
- Timestamp - столбец с метками времени.
- ErrorKey1 - ключевое слово ошибки 1.
- ErrorKey2 - ключевое слово ошибки 2.
- ErrorKey3 - ключевое слово ошибки 3.
- WarningKey1 - ключевое слово предупреждения 1.
- WarningKey2 - ключевое слово предупреждения 2.
- WarningKey3 - ключевое слово предупреждения 3.
- ErrorKey1Color (ff0000)- цвет выделения ошибки 1.
- ErrorKey2Color (ff0000)- цвет выделения ошибки 2.
- ErrorKey3Color (ff0000) - цвет выделения ошибки 3.
- WarningKey1Color (ffff00)- цвет выделения предупреждения 1.
- WarningKey2Color (ffff00)- цвет выделения предупреждения 2.
- WarningKey3Color (ffff00) - цвет выделения предупреждения 3,
- Limit (100) - лимит записей.

- SearchLimit (1000) - лимит поиска.
- PanelVisible (true) - видимость панели управления.

## СОДЕРЖАНИЕ

1. Установка и настройка базы данных MySQL
  - 1.1. Установка MySQL сервера
  - 1.2. Настройка MySQL сервера
  - 1.3. Установка ODBC-драйвера MySQL
  - 1.4. Установка MySQL Workbench
2. создание баз данных, создание таблиц БД, занесение полей
  - 2.1. Последовательность действий
  - 2.2. Настройка компонентов в ZETView

## 1

## 1 УСТАНОВКА И НАСТРОЙКА БАЗЫ ДАННЫХ MYSQL

## 1.1 Установка MySQL сервера

Запустите файл-установщик mysql.exe.

В окне выбора типа установки выберите «Typical» (рисунок 1.1.1), нажмите кнопку «Next».

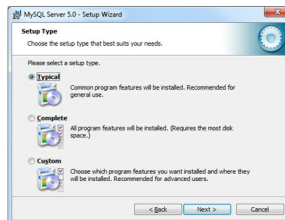


Рисунок 1.1.1 – Окно выбора типа установки MySQL

После окончания установки будет предложено настроить конфигурацию MySQL сервера (рисунок 1.1.2). Оставьте галочку отмеченной и нажмите кнопку «Finish».



Рисунок 1.1.2 – Окно завершения установки MySQL сервера

## 1.2 Настройка MySQL сервера

Окно конфигуратора MySQL сервера (рисунок 1.2.1) откроется сразу после завершения установки.

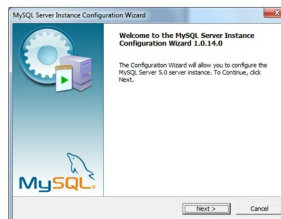


Рисунок 1.2.1 – Главное окно конфигуратора MySQL сервера

Выберите детальную настройку, как это показано на рисунке 1.2.2.

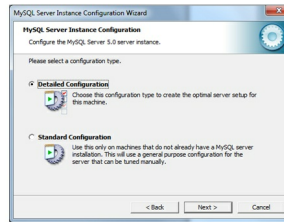


Рисунок 1.2.2 – Выбор детальной настройки MySQL сервера

На всех остальных шагах оставляем настройки по умолчанию, нажимая кнопку «Next», кроме шага выбора кодировки (рисунок 1.2.3). Отмечаем пункт «Manual Selected Default Character Set / Collation» и выбираем из списка кодировку «utf8».

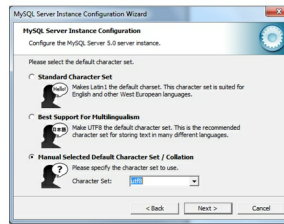


Рисунок 1.2.3 – Окно выбора кодировки по умолчанию MySQL сервера



Рисунок 1.2.4 – Окно ввода пароля пользователя root, в качестве пароля можно ввести символьную или числовую последовательность (исключая специальные символы)

### 1.3 Установка ODBC-драйвера MySQL

Запустите файл-установщик mysql-connector-odbc-5.1.6-win32.msi.

В окне выбора типа установки выберите «Typical» (рисунок 1.3.1), нажмите кнопку «Next».



Рисунок 1.3.1 – Окно выбора типа установки ODBC-драйвера MySQL

## 1.4 Установка MySQL Workbench

Запустите файл-установщик mysql-workbench-oss-5.2.22-rc-win32.msi.

В окне выбора типа установки выберите «Complete» (рисунок 1.4.1), нажмите кнопку «Next».

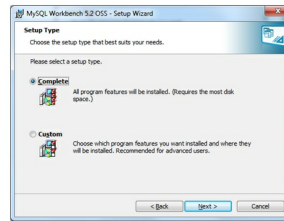


Рисунок 1.4.1 – Окно выбора типа установки MySQL Workbench

## 1.5 создание базы данных, создание таблицы БД, занесение полей

Пример создания базы данных, таблицы и занесения полей различного типа.

Название базы данных: SensorDataBase

Название таблицы: Sensors

Последовательность действий

- Запускаем командную консоль mysql.exe (обычно располагается в меню Пуск ОС Windows)
- В командной строке вводим пароль, указанный при установке MySQL
- Создание базы данных с именем 'SensorDataBase' и кодировкой utf8\_general\_ci

```
CREATE SCHEMA `SensorDataBase` DEFAULT CHARACTER SET `utf8` COLLATE `utf8_general_ci`;
```

- Добавление к базе данных 'SensorDataBase' новой таблицы с именем 'Sensors', ключевым полем с именем '#' и кодировкой utf8\_general\_ci

```
CREATE TABLE `sensordatabase`.`Sensors` (
  `#` INT UNSIGNED NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`#`),
  UNIQUE INDEX `#_UNIQUE` (`#` ASC)
  DEFAULT CHARACTER SET = utf8
  COLLATE = utf8_general_ci;
```

- Добавление поля с именем 'SensorName' в таблицу 'sensors' базы данных 'sensordatabase', которое следует сразу за ключевым полем

```
ALTER TABLE `sensordatabase`.`Sensors` ADD COLUMN `SensorName` VARCHAR(45) NULL AFTER `#`;
```

- Добавление поля с именем 'SensorType' в таблицу 'sensors' базы данных 'sensordatabase', которое следует сразу за полем 'SensorName'

```
ALTER TABLE `sensordatabase`.`Sensors` ADD COLUMN `SensorType` VARCHAR(45) NULL AFTER `SensorName`;
```

- Удаление строкового поля 'SensorType'

```
ALTER TABLE `sensordatabase`.`Sensors` DROP COLUMN `SensorType`;
```

- Добавление поля 'SensorType' типа INT с флагом без знаковый

```
ALTER TABLE `sensordatabase`.`Sensors` ADD COLUMN `SensorType` INT UNSIGNED NULL AFTER `SensorName`;
```

- Добавление нового поля 'TestField' типа INT со всеми возможными флагами

```
NOT NULL - Not Null
UNSIGNED - Unsigned Data Type
ZEROFILL - Fill up values for this column with 0 if it's numeric
аналогичная операция по добавлению полей DOUBLE, FLOAT
```

```
ALTER TABLE `sensordatabase`.`Sensors` ADD COLUMN `TestField` INT UNSIGNED ZEROFILL NOT NULL AFTER `SensorType`;
```

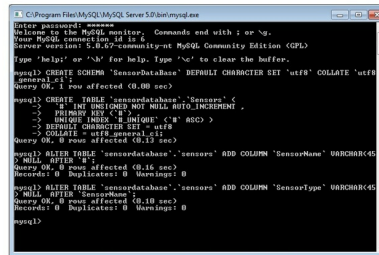


Рисунок 1.5.1 – Примерный вид консоли SQL

**Примечание:**

Для текстового VARCHAR(xx), DATETIME, DATE, TIME, доступен один тип флага NOT NULL, однако, не рекомендуется его использовать, оставая по умолчанию как представлено на примере 'e' и 'f' пунктов.

Более подробную информацию по базам данных можно получить из электронных и печатных изданий по MySQL.

Для моделирования, разработки и поддержки баз данных, удобно пользоваться специализированным программным обеспечением, предоставляющим "обертку", представленную в виде графических пользовательских интерфейсов над командами SQL. Пример такого программного обеспечения MySQL Workbench.

## 1.6 Настройка компонента «База данных» в ZETView









Настраиваемые свойства	
Общие свойства (Базы данных)	
Caption	База данных_1
VisibleInRun	True
m_sHelpString	База данных_1
Координата ЛН угла	
Left	-87
Top	98
Частные свойства	
IP	127.0.0.1
DSN	MySQL
Database	SenserDataBase
User	root
Password	111111
Table	Sensor
Delimiter	
Column	
Timestamp	
ErrorKey1	
ErrorKey2	
ErrorKey3	
WarningKey1	
WarningKey2	
WarningKey3	
ErrorKey1Color	 ff0000
ErrorKey2Color	 ff0000
ErrorKey3Color	 ff0000
WarningKey1Color	 ffff00
WarningKey2Color	 ffff00
WarningKey3Color	 ffff00
Limit	100
SearchLimit	1000
PanelVisible	True

Рисунок 1.6.2 – Настройка параметров компонента «База данных»

Описание свойств:

IP – IP адрес компьютера, на котором установлена база данных.  
 DSN (Data Source Name) – Имя источника данных  
 Database – Название базы данных, к которой необходимо произвести подключение  
 User – пользователь (имя, введенное при установке MySQL)  
 Password – пароль пользователя  
 Table – таблица, к которой производим подключение  
 Delimeter – текстовый разделитель столбцов (полей) базы данных  
 Column – столбец, значения которого будут на выходе  
 Timestamp – столбец с метками времени  
 ErrorKey1 – ключевое слово ошибки 1  
 ErrorKey2 – ключевое слово ошибки 2  
 ErrorKey3 – ключевое слово ошибки 3  
 WarningKey1 – ключевое слово предупреждения 1  
 WarningKey2 – ключевое слово предупреждения 2  
 WarningKey3 – ключевое слово предупреждения 3  
 ErrorKey1Color – цвет выделения ошибки 1  
 ErrorKey2Color – цвет выделения ошибки 2  
 ErrorKey3Color – цвет выделения ошибки  
 WarningKey1Color – цвет выделения предупреждения 1  
 WarningKey2Color – цвет выделения предупреждения 2  
 WarningKey3Color – цвет выделения предупреждения 3  
 Limit – лимит записей  
 SearchLimit – лимит поиска  
 PanelVisible – видимость панели управления.

#### 1.7 Добавление новой записи в «Базу данных» из ZETView

Для добавления новой записи в базу данных, с которой установлено соединение необходимо подать на вход компонента «База данных» текстовую строку в которой символ Delimeter (по умолчанию «») будет разделять данные по столбцам таблицы (по полям).

Простой пример представлен на рисунках 1.7.1 и 1.7.2

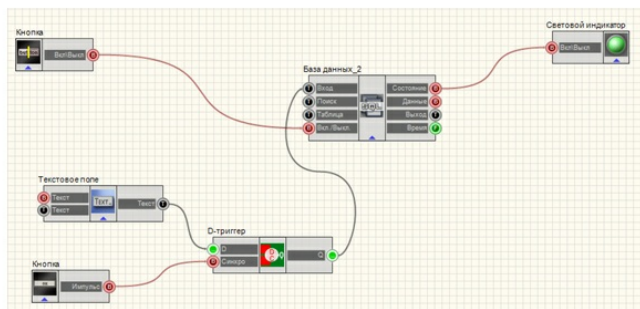


Рисунок 1.7.1 – Добавление новой записи (строки). Вид проектировщика



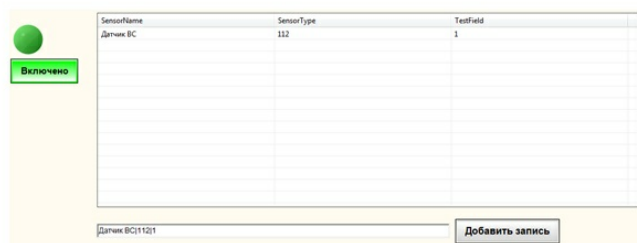


Рисунок 1.7.2 – Добавление новой записи (строки). Вид оператора

Данный проект имеет «текстовое поле», в котором вводим строчку с разделителями. Данная строка попадает на вход «триггера», выполняющего сохранение введенной строки и «кнопки», нажимая на которую, сохраненная в «триггере» строка попадает на вход компонента «база данных» после чего заносится в виде новой записи в базу данных, к которой выполнено подключение.

На примере видно, что в базу данных попала запись о датчике ВС тип 112 с пометкой 1.

## 29.2.Выбор каталога

**Выбор каталога** - компонент. Позволяет оператору выбирать папку на своем компьютере и передать адрес на выходной контакт. Таким образом можно, например, предоставить оператору возможность выбирать папку для автоматического сохранения отчётов.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - контакт, при поступлении управляющего импульса открывает указанный директорию и отправляет данные на выход.
- Синхро - отправить имя директории.

#### ➤ Выходные

- Путь - имя директории.

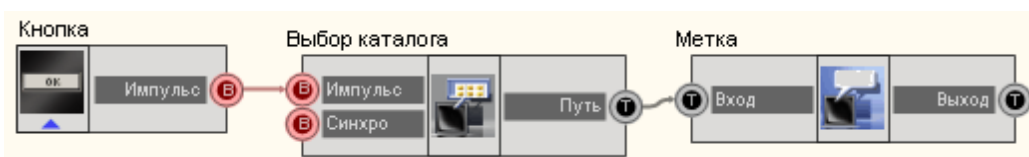
### Настраиваемые свойства:

- Общие свойства (окружение):
  - [Приведены по данной ссылке.](#)<sup>92</sup>
- Частные свойства (в скобках значение, установленное по умолчанию):  
FolderName (C:\)- имя директории



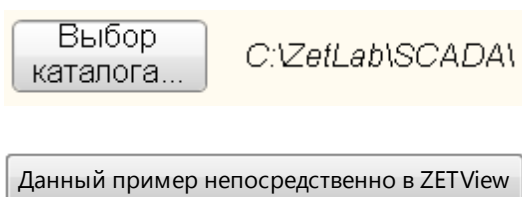
## Пример

### Проект в SCADA ZETView



При нажатии кнопки без фиксации, компонент "Выбор каталога" получает управляющий импульс на вход и предлагает выбрать каталог, затем передает полученное значение с выхода "Путь", в данном примере результат просто отображается в текстовой метке. Обычно этот компонент используется совместно с компонентом "Отчет", для выбора директории сохранения отчетов.

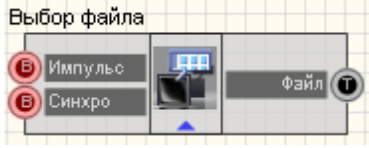
### Результат работы проекта



## 29.3.Выбор файла

**Выбор файла** - компонент. Предназначен для выбора оператором файла, передает путь файла.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - контакт, при поступлении управляющего импульса открывает проводник для выбора файла.
- Синхро - контакт, при поступлении управляющего импульса передает путь к файлу на выход компонента.

#### ➤ Выходные:

- Файл - контакт, с которого поступает путь к выбранному файлу.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства:

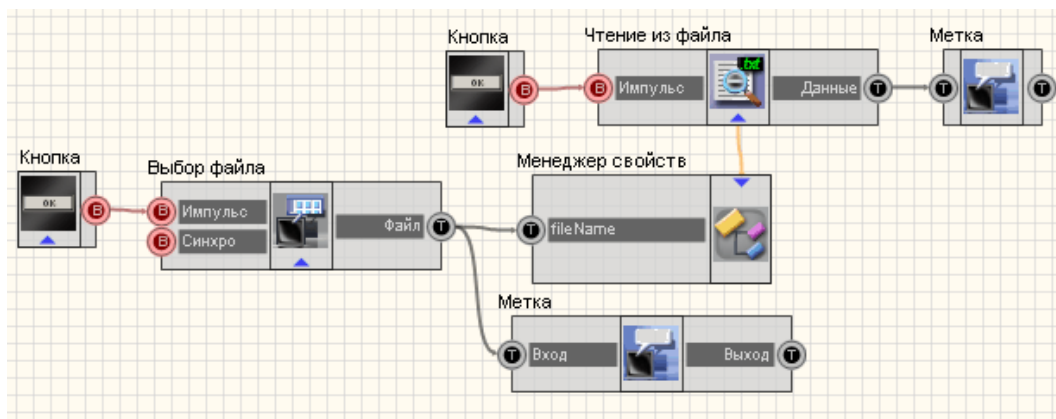
- FileName - имя файла.



## Пример

### Проект в SCADA ZETView

**!** В данном примере используется произвольный txt-файл не включенный в состав примера. При запуске примера, чтобы увидеть результат его работы, необходимо выбрать любой txt-файл на компьютере с помощью кнопки "Орен", путь к файлу отобразится ниже. А затем нажать кнопку "Прочитать файл". Результат может отличаться от представленного в данном примере в зависимости от содержимого выбранного txt-файла.



В данной схеме показана работа компонента [Выбор файла](#)<sup>[592]</sup>.

При нажатии [Кнопки без фиксации](#)<sup>[367]</sup> посылается импульс на вход компонента [Выбор файла](#)<sup>[592]</sup>, вследствие чего открывается проводник для выбора файла. Путь к выбранному файлу отображается с помощью [Метки](#)<sup>[554]</sup>. Также, строка, содержащая путь к файлу, поступает на [Менеджер свойств](#)<sup>[534]</sup>, подключенный к компоненту [Чтение из файла](#)<sup>[607]</sup>. При получении управляющего импульса с [Кнопки без фиксации](#)<sup>[367]</sup>, данный компонент производит чтение содержимого выбранного файла и выводит результат с помощью [Метки](#)<sup>[554]</sup>.

Для данного примера можно использовать любой txt-файл.

### Результат работы проекта

Открыть проводник:

Open

Выбранный файл:

C:\MyFile.txt

Прочитать файл

Содержимое файла:

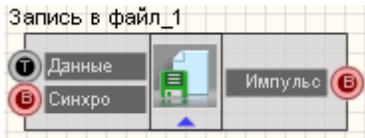
Hello World!

Посмотреть пример непосредственно в ZETView

## 29.4.Запись в файл

**Запись в файл** - компонент. Позволяет записывать данные в файл при работе проекта. Таким образом возможно использование динамических данных не связанных с работой проекта.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤Входные:

- Данные - данные для записи в файл.
- Синхро - команда на запись в файл.

#### ➤Выходные

- Импульс - информирует об окончании записи в файл.

### Настраиваемые свойства:

#### ➤Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

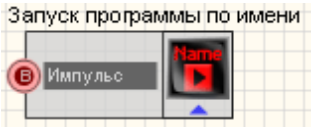
#### ➤Частные свойства (в скобках значение, установленное по умолчанию):

- fileName - имя файла для записи.
- turemode (По изменению) - режим работы:
  1. По изменению.
  2. По импульсу.
- writemode (Перезаписывать) - режим записи:
  1. Перезаписывать.
  2. Добавить в конец.

## 29.5.Запуск программы по имени

**Запуск программы по имени** - компонент. Позволяет запускать программы (или открывать файлы) непосредственно из проекта.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - при получении управляющего импульса на данный вход, компонент запускает указанный файл.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

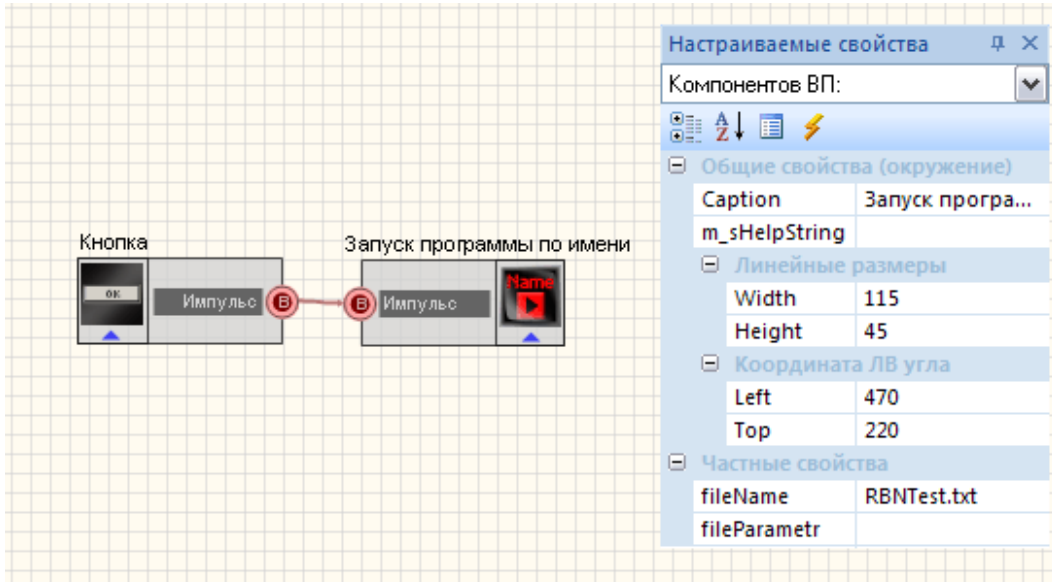
#### ➤ Частные свойства:

- fileName - имя файла для запуска.
- fileParametr - параметр для запускаемого файла.



## Пример

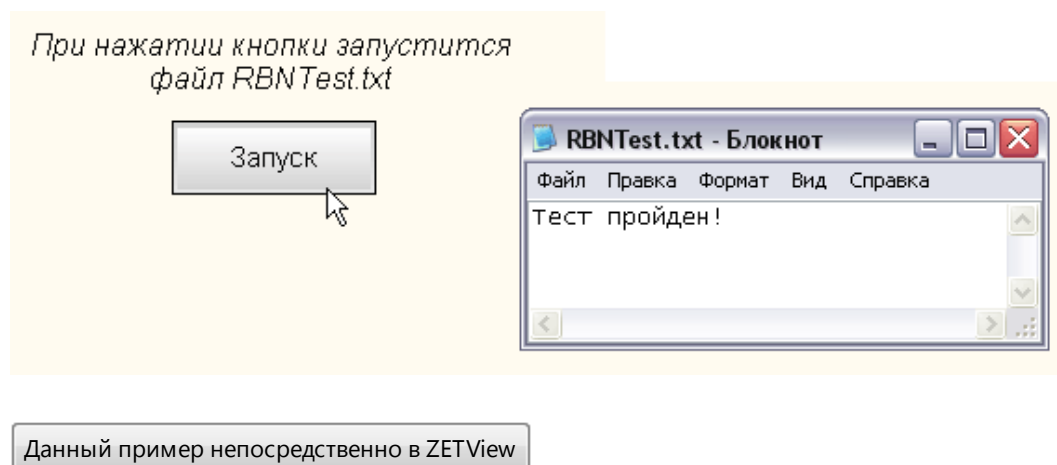
### Проект в SCADA ZETView



The screenshot shows a SCADA ZETView project interface. On the left, a button labeled 'Кнопка' with 'Импульс' and a red 'E' icon is connected to a component labeled 'Запуск программы по имени' which also has 'Импульс' and 'Name' labels. On the right, a 'Настраиваемые свойства' (Properties) panel is open, showing the following settings:

Настраиваемые свойства	
Компонентов ВП:	
Общие свойства (окружение)	
Caption	Запуск програ...
m_sHelpString	
Линейные размеры	
Width	115
Height	45
Координата ЛВ угла	
Left	470
Top	220
Частные свойства	
fileName	RBNTTest.txt
fileParametr	

### Результат работы проекта



## 29.6.Отчет

**Отчет** - компонент. Предназначен для записи различных величин в файл по пользовательскому шаблону.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- **Значение** - запись значения в отчет. К этому контакту подключаются компоненты, значения которых требуется внести в отчет. Значения накапливаются до момента сохранения отчета, таким образом, в файл записывается последние значения, пришедшие от каждого подключенного компонента. Контакт может образовывать множественные соединения. Интерфейс контакта "Любой тип соединений"
- **Показ** - При получении значения "1" этим контактом, отображаются текущие значения.
- **Имя** - При установленном флаге "Автоназвание" в окне настроек отчета, созданному файлу при сохранении в качестве имени присваивается значение,

пришедшее на этот вход.

- Импульс - При получении значения "1" этим контактом, накопленные значения сохраняются в файл.
- Показ - При получении значения "1" этим контактом, открывается файл последнего сохраненного отчета.

➤ Выходные:

- Импульс - данный контакт рассылает значение "1" подключенным компонентам по окончанию формирования отчета. Время формирования отчёта зависит от объёма данных, которые необходимо записать в файл. Кроме того, в большинстве проектов отчеты создаются автоматически. Таким образом, для повышения информативности, в проект может быть добавлен индикатор создания отчета.

### Настраиваемые свойства:

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

➤ Частные свойства:

- FileTemplateName - имя файла-шаблона для отчёта. Расширение файла шаблона должно совпадать с расширением будущего отчета. Т.е. если выставить свойство reporttype: " Microsoft Word (.doc).", то соответственно файл шаблона отчета должен иметь расширение \*.doc.
- Show (false) - показать отчёт после его формирования.
- Replace (false)- заменить неиспользуемые идентификаторы в отчёте. Так как в шаблонах используется форма записи данных типа &1\_Massiv1/& (потом подобные надписи заменяется значениями записанными в отчет), то, если например, в файл шаблона рассчитан на пять значений , а получит четыре, то оставшееся незаполненное значение будет заменено прочерком. Если свойство Replace установить "False", то лишние строки заменятся не будут. См.рис.

Так выглядит файл  
шаблона отчета, в  
формате \*.doc

1	&1_massiv1/&
2	&2_massiv1/&
3	&3_massiv1/&
4	&4_massiv1/&
5	&5_massiv1/&

Так выглядит файл  
отчета, в формате \*.doc  
если свойство Replace  
установлено True

1	23,86
2	13,55
3	14,18
4	27,11
5	-

Так выглядит файл  
отчета, в формате \*.doc  
если свойство Replace  
установлено False

1	23,86
2	13,55
3	14,18
4	27,11
5	&5_massiv1/&

- Reset (false) - обнуление базы зафиксированных значений. Данное свойство используется когда файл отчета необходимо перезаписать. Например, при первой записи в отчет было записано 40 значений, после отчет переформировали в тот же самый файл, записав в него 20 значений. Если при формировании отчета значение Reset было установлено "True", ZETView сотрет все предыдущие значения и запишет на их место новые. Если значение свойства выставлено "False" то данные будут



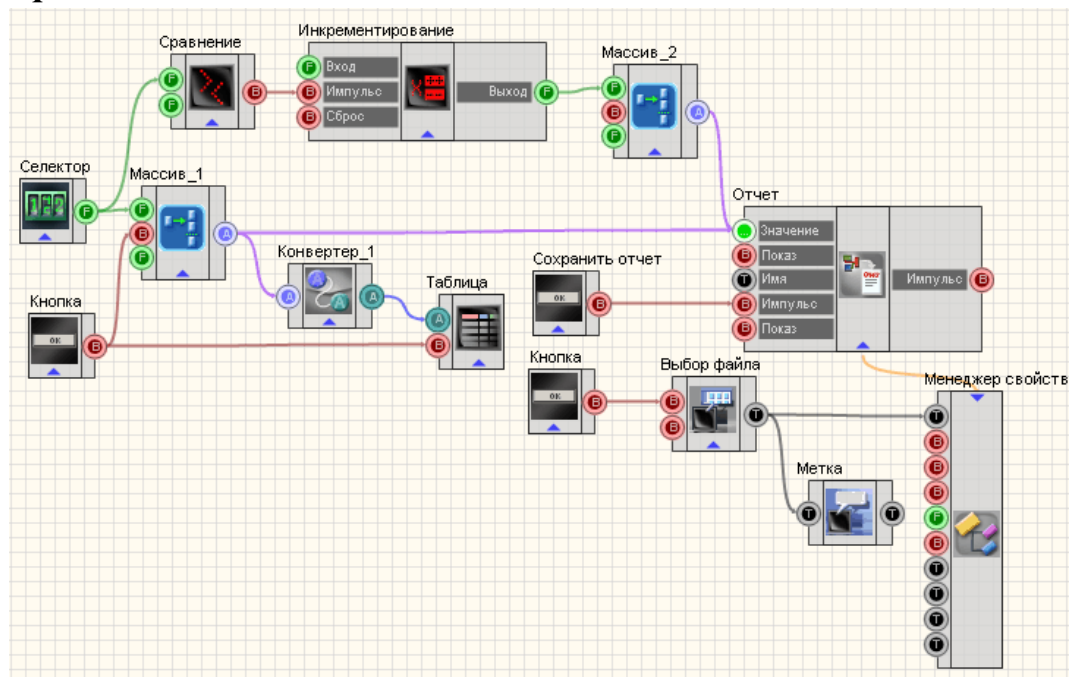
просто заменены, результаты значения которых не менялись останутся нетронутыми.

- Precision (0) - количество знаков после запятой в отчёт значений с плавающей запятой.
- AutoName (false) - автоматическое присваивание имени отчёта.
- FolderName - имя директории, куда будут помещаться автоматически сформированные отчёты.
- reporttype (Блокнот (.txt))- тип отчёта:
  1. Блокнот (.txt).
  2. Microsoft Word (.doc).
  3. Microsoft Excel (.xls).
  4. Файл результатов (.dtu)
- BufferVar - метка вставки информации из буфера обмена в отчёт при формировки отчёта.
- delimiter (Запятая) - разделитель целой и дробной части.
  1. Запятая.
  2. Точка.

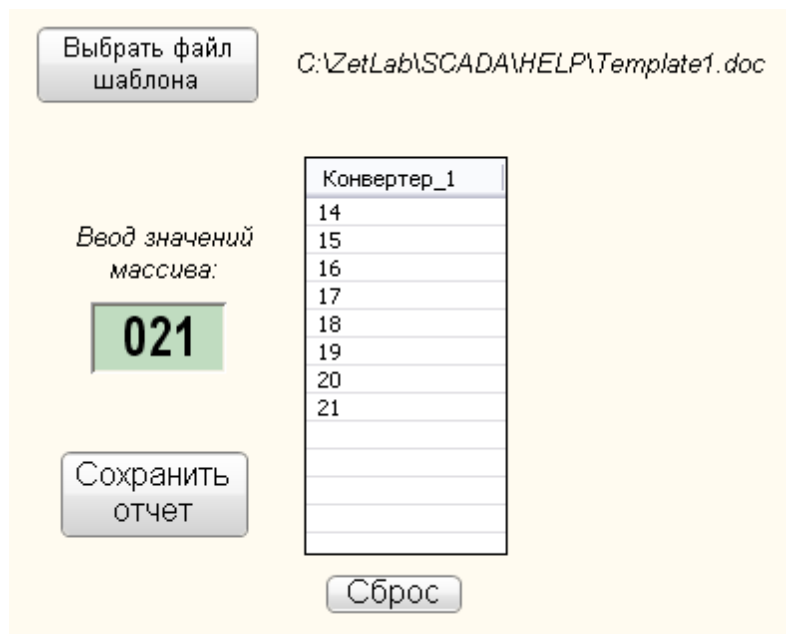


## Пример

### Проект в SCADA ZETView



### Результат работы проекта



## 29.7. Поиск файлов в директории

**Поиск файлов в директории** - компонент. Позволяет находить файлы в заданной директории, по расширению, указанному в маске.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

**Параметры:**

➤ Входные:

- Путь - название директории.
- Маска - маска для поиска. Вход, свойство FileExtension. На этот вход подается текстовая информация о расширении файлов которые необходимо найти. В формате: "\*txt" или "\*.txt", без кавычек.

➤ Выходные

- Файлы - выход с которого поступает строка, в которой перечисляются найденные файлы (с разделителем ;), указывается не просто имя файла, а абсолютный путь к каждому найденному файлу.

**Настраиваемые свойства:**

➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

➤ Частные свойства (в скобках значение, установленное по умолчанию):

- DirectionName (c:\) - название директории в которой будет производиться поиск файлов.
- FileExtension (\*.txt) - расширение файлов, поиск которых будет производиться. Указывается в формате \*.txt или \*.txt.
- AddFolderName (true) - добавлять имя папки к имени файла.

**Пример****Проект в SCADA ZETView**

В этом проекте пользователь выбирает каталог и расширение разыскиваемых файлов, после чего на экран выводятся все найденные по маске файлы в указанном каталоге. Выбор каталога осуществляется путем нажатия кнопки без фиксации с надписью "Выбор каталога". Кнопка передает управляющий импульс компоненту "Выбор каталога". При нажатии кнопки откроется стандартный проводник Windows в отдельном окне, и пользователю будет предложено выбрать каталог для поиска. Когда каталог выбран нажать кнопку "Ок" проводника. Тогда путь к выбранному каталогу появится в текстовой метке. Как видно из схемы, полученный компонентом "Выбор каталога" путь передается далее, в виде текстовой строки на компонент "Метка", который отображает выбранный пользователем путь, и на компонент "Поиск файлов в директории", указывая в какой директории производить поиск. Также к компоненту "Поиск файлов в директории" подключен комбинированный список, позволяющий пользователю выбрать необходимое расширение файла из списка. Результат поиска выводится на текстовую метку.

**Результат работы проекта**

Выбор каталога...

C:\WINDOWS\

Искать файлы с расширением:

\*txt

Результат:

C:\WINDOWS\OEWABLog.txt;C:\WINDOWS\SchedLgU.Txt;

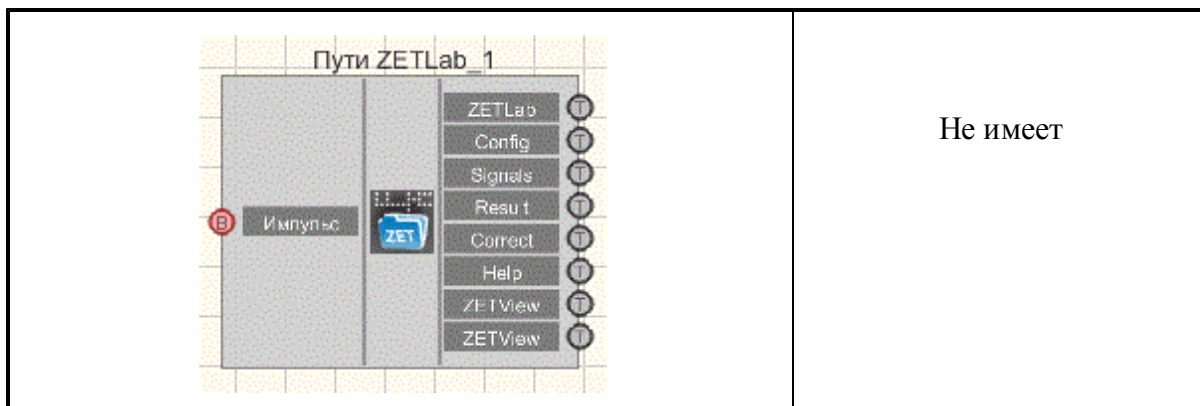
Данный пример непосредственно в ZETView

## 29.8. Пути ZETLab

**Пути ZETLab** - компонент. Позволяет устанавливать корневую директорию в ZETLab, директории содержащие файлы записанных сигналов, результаты обработки, файлы справки и т.д. данный компонент нужен для удобства использования стандартных директорий ZETLab, используется при работе с отчетами, для работы с файлами, когда, например, какие-либо данные сохраняются в директории по умолчанию.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------

**Параметры:**

## ➤ Входящие:

- Импульс - импульс на чтение путей ZETLab

## ➤ Выходные

- ZETLab - корневая директория ZETLab.
- Config - путь к папке с текущим списком устройств.
- Signals - путь к папке с записанными сигналами.
- Result - путь к папке, где хранятся результаты обработки.
- Correct - путь к папке, содержащей пользовательские поправки.
- Help - путь к файлам справки ZETLab.
- ZETView - директория ZETView.
- ZETView - справка ZETView

**Настраиваемые свойства:**

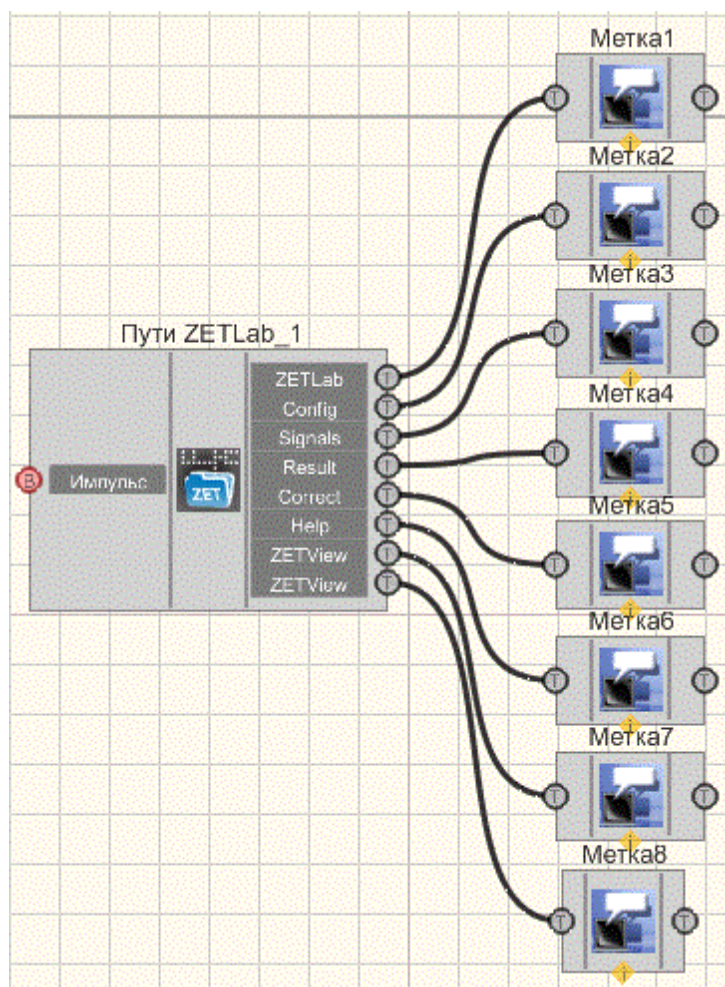
## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>92</sup>

## ➤ Частные свойства:

Не имеет.

***Пример******Проект в SCADA ZETView***



*Результат работы проекта*



*Корневая директория ZETLab:*

C:\Program Files\ZETLab\

*Список устройств:*

C:\Users\trebushkova\Documents\ZETLab\config\

*Сигналы:*

C:\Users\Public\Documents\ZETLab\Trebushkova\signals\s120523\_130734\s120806\_135039\

*Результаты обработки:*

C:\Users\Public\Documents\ZETLab\Trebushkova\result\s120523\_130734\s120806\_135039\

*Пользовательские поправки:*

C:\ProgramData\ZETLab\correct\

*Файлы справки ZetLab:*

C:\Program Files\ZETLab\hlp\

*Директория ZETView:*

C:\Program Files\ZETLab\scada\

*Справка ZETView:*

C:\Program Files\ZETLab\scada\HELP\

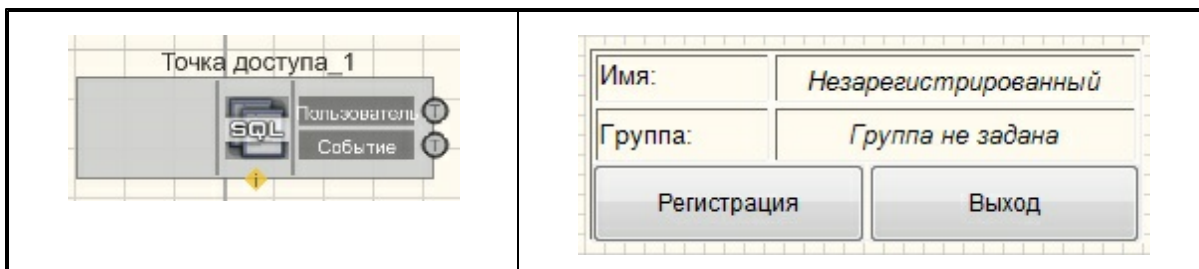
Данный пример непосредственно в ZETView

## 29.9.Точка доступа

**Точка доступа** - компонент. Компонент предназначен для выбора текущего пользователя, который является, например, оператором, диспетчером, инженером с последующим разграничением прав доступа, реализованных вне данного компонента. Имя выбранного пользователя, так же может заноситься при внесении сообщений в базу данных, при квитировании различных информационных сообщений сложных систем, либо заноситься в отчеты, создаваемые в результате обычной каждодневной операционной работы оператора.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



### Параметры:

#### ➤ Выходные

- Пользователь - устанавливаются данные о пользователе;
- Событие - устанавливаются данные о событии.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#) <sup>92</sup>

#### ➤ Частные свойства:

- IP - IP адрес компьютера, на котором установлена база данных.
- DSN - имя источника данных DSN.
- Database - название базы данных, к которой необходимо произвести подключение.
- User - пользователь (имя, введенное при установке MySQL).
- Password - пароль пользователя.
- Table - таблица, к которой производим подключение.
- Delimiter - текстовый разделитель столбцов (полей) базы данных.

Компонент основан на применении базы данных MySQL. *База данных MySQL* настраивается на локальной или удаленной машине со следующими параметрами:

- ✓ Количество полей – 4, первое поле целого типа INT, остальные символьные VARCHAR(64).
- ✓ Поле №1 – ключевое индексное поле,
- ✓ Поле №2 – Группа, к которой принадлежит пользователь (возможно должность),
- ✓ Поле №3 – Имя пользователя,
- ✓ Поле №4 – Пароль пользователя.

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI
#	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Имя	VARCHAR(64)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Группа	VARCHAR(64)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Пароль	VARCHAR(64)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Пояснение к настройкам: PK – флаг принадлежности к главному ключу; NN – не нулевое поле, обязательно должно быть заполнено; UQ – флаг уникальности индекса; UN – флаг того, что значение является без знаковым целым типом; AI – автоинкрементное.



Настройка баз данных данным документе не рассматривается. Как правило, настройка баз данных производится администратором. Базу данных *MYSQL* можно настроить в консольном режиме с указанием специализированных команд в синтаксисе *SQL*, либо специализированной графической оболочке, в которой большинство действий реализуется с помощью графического пользовательского интерфейса. Примером такого по может служить *MySQL Workbench*.

#	Имя	Группа	Пароль
1	Иванов А.А.	Оператор	*****
2	Ковалев А.И.	Инженер АСУ ТП	*****
3	Устинов А.Ю.	Инженер АСУ ТП	*****
▶*	NULL	NULL	NULL

Для корректной работы компонента необходимо завести базу Данных и настроить сам компонент на доступ к ней путем указания следующих настроек:

IP – Адрес компьютера, если является локальным компьютером, допускается вводить localhost или указать 127.0.0.1

DSN – название источника Базы данных

Database – название базы данных

User – имя администратора базы данных

Password – пароль администратора базы данных

Table – название таблицы, в которой перечислены имена пользователей с указанием Группы (должности) и пароля.

В случае отсутствия базы данных всегда будет использоваться Незарегистрированный пользователь без указания Группы.

По нажатию на кнопку **Регистрация** утанавливается соединение с Базой данных, указанной в настройках компонента. При

нажатию на кнопку **Выход** произойдет переход к незарегистрированному пользователю автоматически, без необходимости выбора данного пункта из таблицы.

При выборе пользователя потребуется ввести его пароль. Зеленым цветом указывается выбранный на текущий момент пользователь. Синим – курсор.

## 29.10. Чтение из файла

**Чтение из файла** - компонент. Предназначен для чтения данных из указанного txt-файла и вывода их в качестве строки.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Импульс - контакт, при поступлении управляющего импульса открывает указанный файл и производит чтение данных, затем отправляет данные на выход.

#### ➤ Выходные:

- Данные - контакт с которого поступает текстовая строка, содержимое указанного текстового файла.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

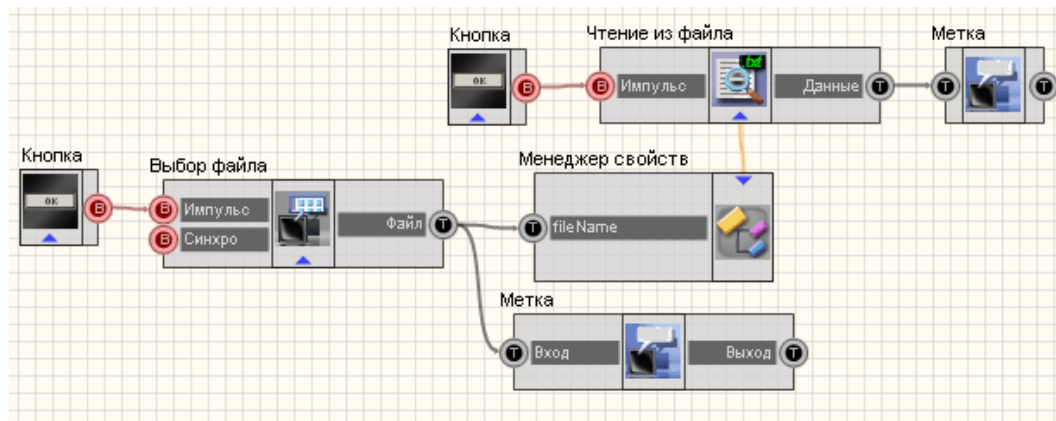
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- FileName - имя файла.
- codename (UTF-8) - кодировка (UTF-8/ASCII)



## Пример

### Проект в SCADA ZETView



В данной схеме показана работа компонента [Чтение из файла](#)<sup>[607]</sup>.

При нажатии [Кнопки без фиксации](#)<sup>[367]</sup> посылается импульс на вход

компонента [Выбор файла](#)<sup>[592]</sup>, вследствие чего открывается проводник для выбора файла. Путь к выбранному файлу отображается с помощью [Метки](#)<sup>[554]</sup>. Также, строка, содержащая путь к файлу, поступает на [Менеджер свойств](#)<sup>[534]</sup>, подключенный к компоненту [Чтение из файла](#)<sup>[607]</sup>. При получении управляющего импульса с [Кнопки без фиксации](#)<sup>[367]</sup>, данный компонент производит чтение содержимого выбранного файла и выводит результат с помощью [Метки](#)<sup>[554]</sup>.

Для данного примера можно использовать любой txt-файл.

### **Результат работы проекта**

Открыть проводник:

Open

Выбранный файл:

C:\MyFile.txt

Прочитать файл

Содержимое файла:

Hello World!

Посмотреть пример непосредственно в ZETView

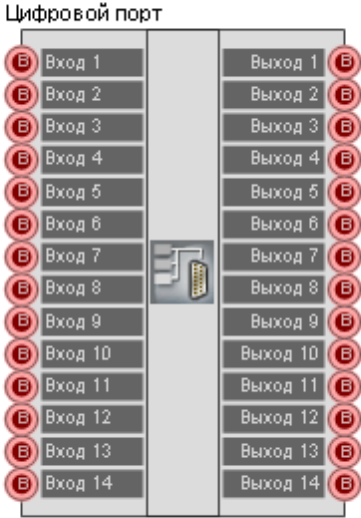
## **Глава 30.Цифровой порт**

### **30.1.Цифровой порт**

**Цифровой порт** - компонент. Позволяет работать с дискретными входными и выходными сигналами.

**!** Для того, чтобы Вход X принимал и реагировал на цифровой сигнал как вход, необходимо в ZETLab в группе программ Сервисные выбрать пункт Диспетчер устройств и в нем установить тип порта Вход или Выход (на прием или на вывод).

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
<p style="text-align: center;">Цифровой порт</p> 	Не имеет

**Параметры:**

## ➤ Входные:

- Вход 1 - входной канал, на который поступает дискретный сигнал.
- ...
- Вход 14 - входной канал, на который поступает дискретный сигнал.

## ➤ Выходные

- Выход 1 - выходной канал, на котором в зависимости от входного сигнала появляется высокий или низкий логические уровни.
- ...
- Выход 14 - выходной канал, на котором в зависимости от входного сигнала появляется высокий или низкий логические уровни.

**Настраиваемые свойства:**

## ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

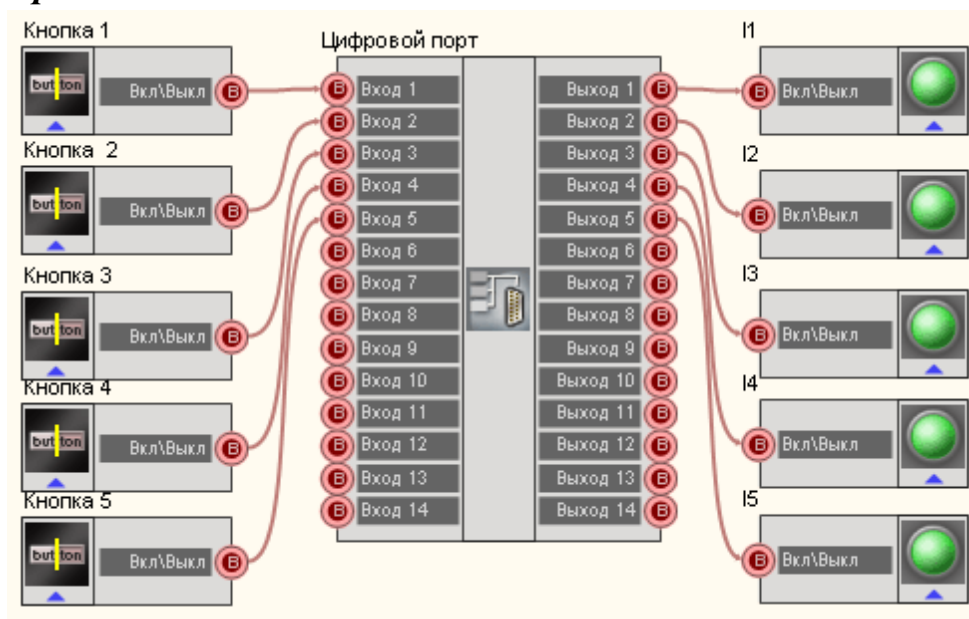
## ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Device (0) - номер устройства.
- OutputMask (65535) - маска вывода цифрового порта.



## Пример

### Проект в SCADA ZETView



В этой схеме компонент [Цифровой порт](#)<sup>[609]</sup> является мостом между цифровым входом и выходом. [Кнопка](#)<sup>[366]</sup> позволяет управлять состоянием входа. [Световой индикатор](#)<sup>[356]</sup> указывает на состояние входа.

### Результат работы проекта



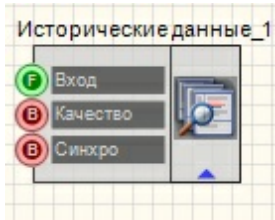
Данный пример непосредственно в ZETView

## Глава 31. OPC

### 31.1. Исторические данные

**Исторические данные** - компонент. Предназначен для предоставления данных OPC клиентам с устройств серии ZET70XX.

#### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

#### Параметры:

##### ➤ Входные:

- Вход - добавление значения в Исторический сервер.
- Качество - метка качества добавляемого значения.
- Синхро - записать данные.

#### Настраиваемые свойства:

##### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

##### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- IP (127.0.0.1) - IP адрес сервера.
- DSN - имя источника данных DSN.
- Database - название базы данных, к которой необходимо произвести подключение.
- User - пользователь (имя, введенное при установке MySQL).
- Password - пароль пользователя.
- Table - таблица, к которой производим подключение.
- type (Авто) - тип синхронизации:
  1. Авто
  2. Вручную.

## 31.2.Обмен данными с OPC

**Обмен данными OPC** - компонент. Предназначен для подключения контроллеров и модулей преобразователей сторонних производителей по OPC технологии клиент-сервер;

Позволяет передавать данные и качество одновременно на несколько OPC-серверов в разные теги

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входящие:

- Тег - данные для записи в тег.
- Качество- метка качества входных данных.

#### ➤ Выходные

- Выход - чтение тега.
- Качество - метка качества выходных данных.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

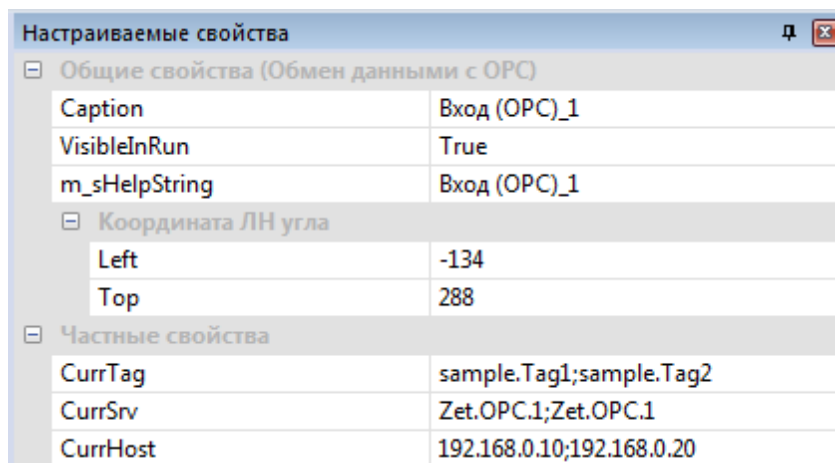
- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства:

- CurrTag - название тега.
- CurrSrv - название сервера.
- CurrHost - название хоста.

В частных свойствах можно указать одновременно несколько хостов, серверов и тегов (число параметров должно быть одинаковым) через точку с запятой.

Пример. Здесь данные кладутся на OPC сервер ZET на хостах 192.168.0.10 и 192.168.0.20 в теги sample.Tag1 и sample.Tag2



Для демонстрации работы сервера ModbusOPC достаточно:

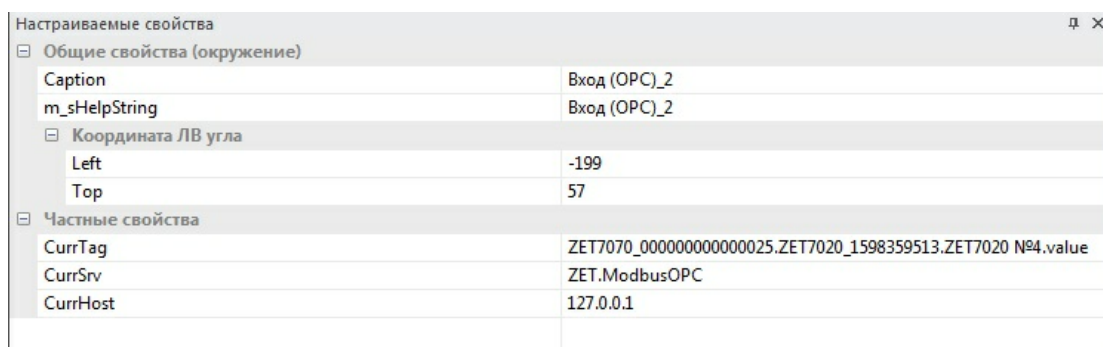
- 1) Подключить устройство 70XX к компьютеру.
- 2) Запустить SCADA-систему ZETView на другом компьютере и поместить на форму компонент «Обмен данными с OPC»
- 2) Открыть окно свойств компонента и ввести параметры:

- CurrTag – полный путь до OPC-тега.

Например: ZET7070\_000000000000273.ZET7021\_268764889.Температура1.value

- CurrSrv – программный идентификатор OPC-сервера – ZET.ModbusOPC
- CurrHost – ввести имя удаленного компьютера, к которому подключено устройство.

Например: 127.0.0.1



- 3) Подключить необходимые элементы к выходу компонента «Обмен данными с OPC» (например, цифровой индикатор, формировщик массивов, световой индикатор и т.д.).
- 4) После запуска проекта ZETView произойдет подключение к OPC-серверу и



запустится обмен данными.



### Пример

#### Проект в SCADA ZETView



#### Результат работы проекта



Данный пример непосредственно в ZETView

## Глава 32. Excel

### 32.1. Документ Excel

**Документ Excel** - компонент, используется для подключения Excel-документа к проекту, на выходе передает указатель на Excel-файл со специального контакта.

**Внешний вид компонента:**

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Вкл\Выкл - в зависимости от логического уровня на данном контакте, компонент подключается/отключается к заданному Excel-файлу.
- Название - контакт, получающий название документа, к которому подключаться.
- Вкл\Выкл - в зависимости от логического уровня на данном контакте, включается либо отключается отображение редактируемого Excel-файла.
- Сохранить - при поступлении управляющего импульса на данный контакт, сохраняет подключенный документ.

#### ➤ Выходные

- Книга - контакт, с которого поступает указатель на Excel-файл.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

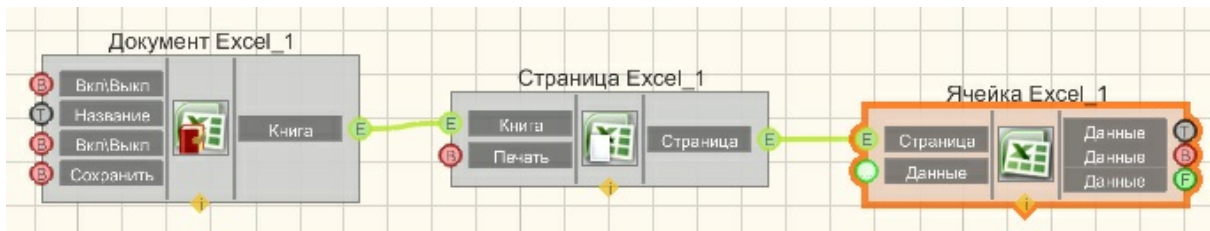
- [Приведены по данной ссылке.](#)<sup>[92]</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- ExcelDocFileName - название документа, к которому подключаемся, если не указано, создается чистый документ.
- ExcelDocVisible (false) - параметр, устанавливающий видимость редактируемого excel-файла. Принимает значения true/false. Если значение параметра = true, ZETView обратится к MSExcel и откроет указанный файл.
- Password\_read - строка, содержащая пароль на открытие указанного Excel-файла.
- Password\_write - строка, содержащая пароль на запись данных в указанный Excel-файл.

### Описание:

Для записи данных в Excel-файл, используется совместно с компонентами "[Страница Excel](#)"<sup>[618]</sup> и "[Ячейка Excel](#)"<sup>[620]</sup>. Соединять компоненты нужно в следующей последовательности:



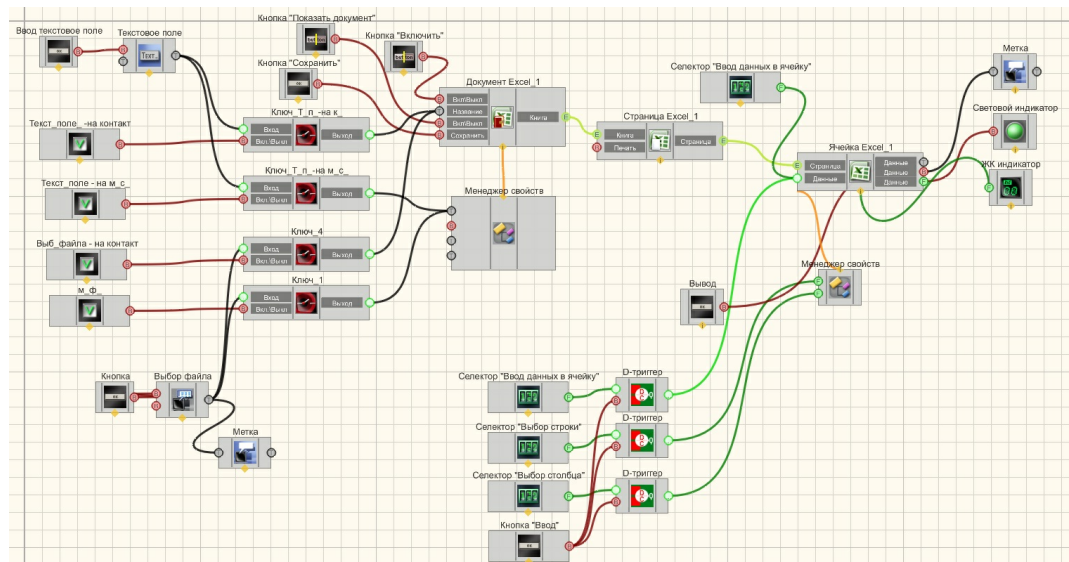
Данный компонент скорее является составным блоком, нежели самостоятельным компонентом. Необходим при работе с excel-файлами. Несмотря на то, что компоненты "[Документ Excel](#)"<sup>[615]</sup>, "[Страница Excel](#)"<sup>[618]</sup> и "[Ячейка Excel](#)"<sup>[620]</sup> имеют входные и выходные контакты одного типа, с этих контактов поступают различающиеся данные. С компонента "[Документ Excel](#)"<sup>[615]</sup> поступает указатель на книгу, а с компонента "[Страница Excel](#)"<sup>[618]</sup> указатель на страницу. Поэтому нельзя соединить "[Документ Excel](#)"<sup>[615]</sup> и "[Ячейка Excel](#)"<sup>[620]</sup> напрямую.

Компонент стабильно работает с версией Microsoft Office Excel 2007.

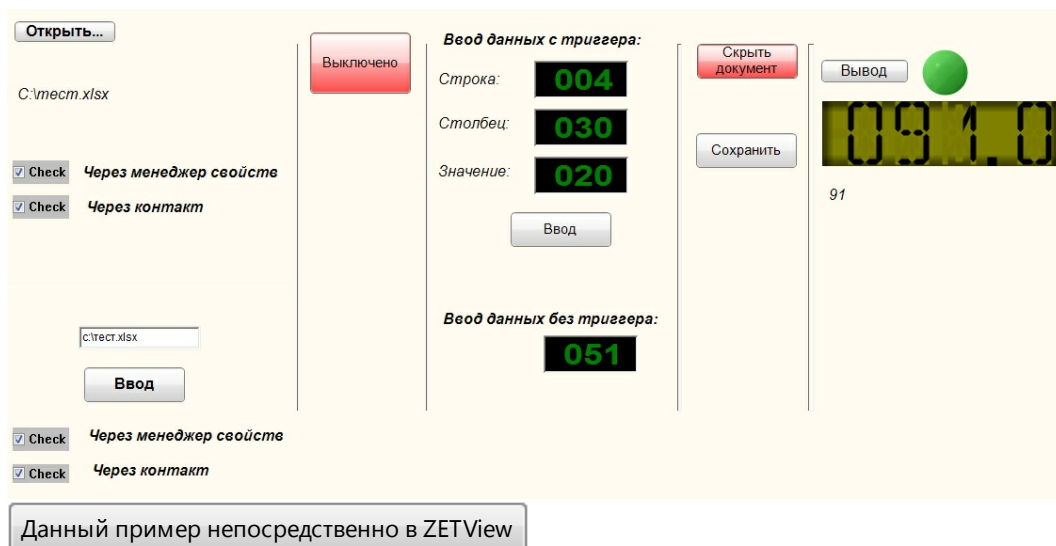


## Пример

### Проект в SCADA ZETView



### Результат работы проекта



## 32.2.Страница Excel

**Страница Excel** - компонент, используется для получения указателя на необходимую страницу Excel-файла, так же имеет функцию печати выбранной страницы.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
	Не имеет

### Параметры:

#### ➤ Входные:

- Книга - входной контакт. Получает указатель на Excel-файл.
- Печать - входной контакт. При получении управляющего импульса, отправляет указанную страницу на печать. Используется принтер, выбранный в системе по умолчанию.

#### ➤ Выходные

- Страница - выходной контакт. Передает указатель на страницу Excel.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

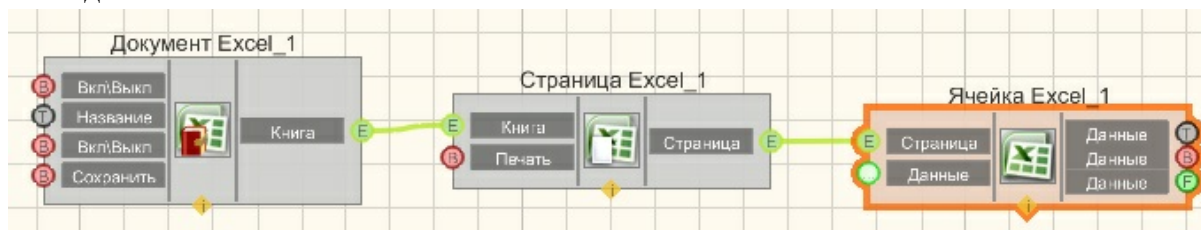
- [Приведены по данной ссылке.](#)<sup>92</sup>

#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- SheetNumber (1) - номер страницы.
- Print\_Copies (0) - количество копий.

### Описание:

Для записи данных в Excel-файл, используется совместно с компонентами ["Документ Excel"](#)<sup>[615]</sup> и ["Ячейка Excel"](#)<sup>[620]</sup>. Соединять компоненты нужно в следующей последовательности:



Данный компонент скорее является составным блоком, нежели самостоятельным компонентом. Необходим при работе с Excel-файлами. Несмотря на то, что компоненты ["Документ Excel"](#)<sup>[615]</sup>, ["Страница Excel"](#)<sup>[618]</sup> и ["Ячейка Excel"](#)<sup>[620]</sup> имеют входные и выходные контакты одного типа, с этих контактов поступают различающиеся данные. С компонента ["Документ Excel"](#)<sup>[615]</sup> поступает указатель на книгу, а с компонента ["Страница Excel"](#)<sup>[618]</sup> указатель на страницу. Поэтому нельзя соединить ["Документ Excel"](#)<sup>[615]</sup> и ["Ячейка Excel"](#)<sup>[620]</sup> напрямую.

Также может использоваться для печати Excel-файла. Для этого достаточна конструкция :

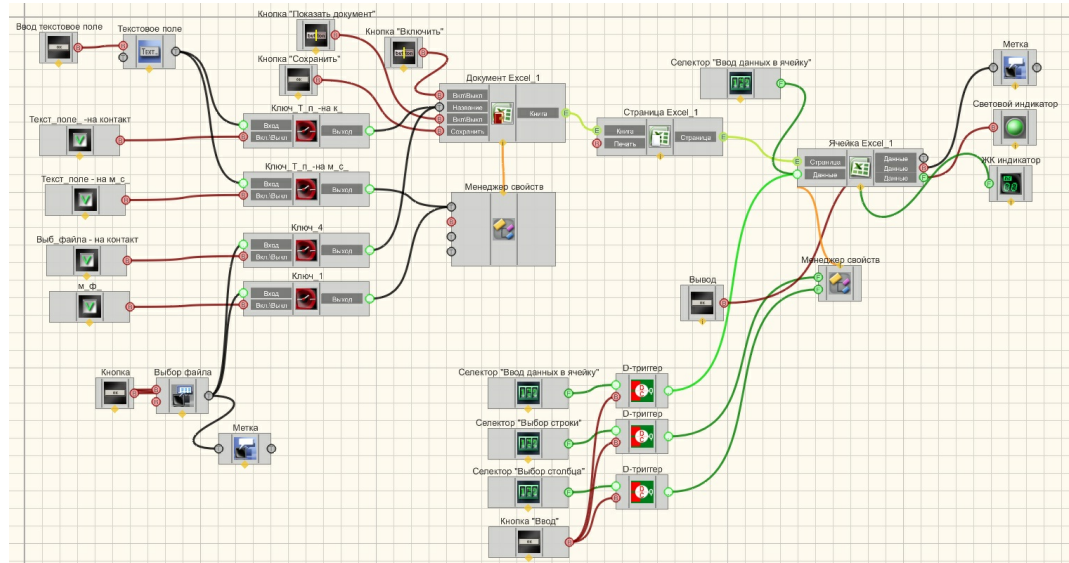
["Документ Excel"](#)<sup>[615]</sup>["Страница Excel"](#)<sup>[618]</sup>

Компонент стабильно работает с версией Microsoft Office Excel 2007.

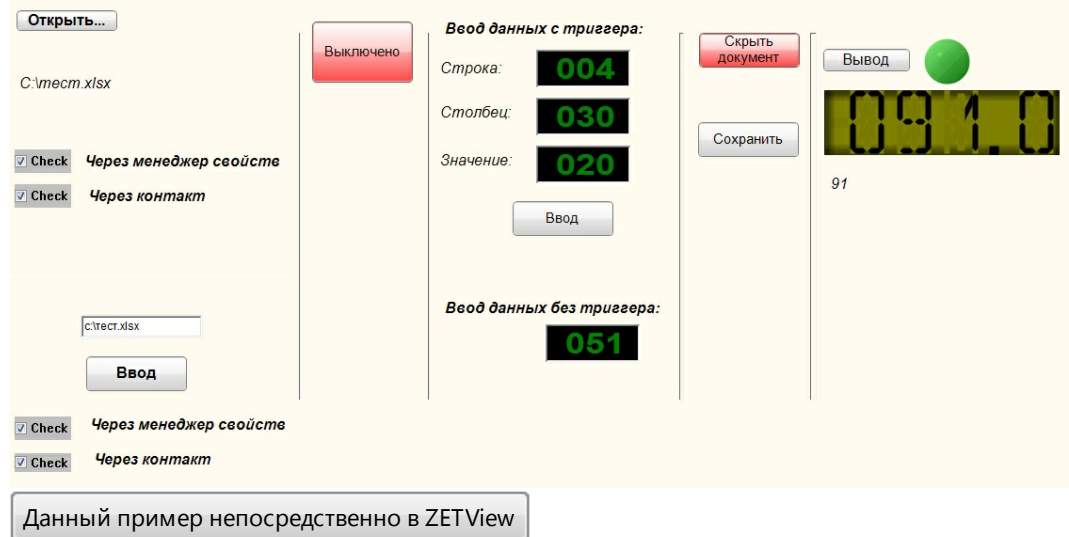


### Пример

### Проект в SCADA ZETView



### Результат работы проекта

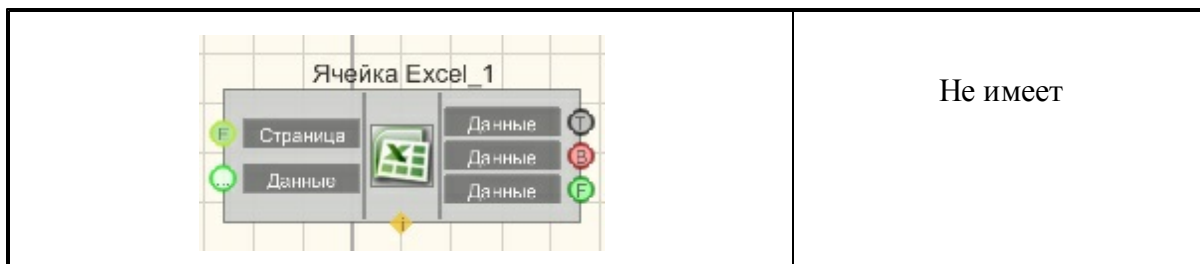


## 32.3.Ячейка Excel

**Ячейка Excel** - компонент, используется для записи данных в указанную ячейку.

### Внешний вид компонента:

Режим проектировщика	Режим оператора
----------------------	-----------------



Не имеет

### Параметры:

#### ➤ Входные:

- Страница - контакт, на который приходит указатель на страницу Excel, в ячейки которой будет производиться запись значений.
- Данные - входной контакт, на который поступают данные любого типа, которые нужно записать в указанную ячейку.

#### ➤ Выходные

- Не имеет.

### Настраиваемые свойства:

#### ➤ Общие свойства (окружение):

- [Приведены по данной ссылке.](#)<sup>[92]</sup>

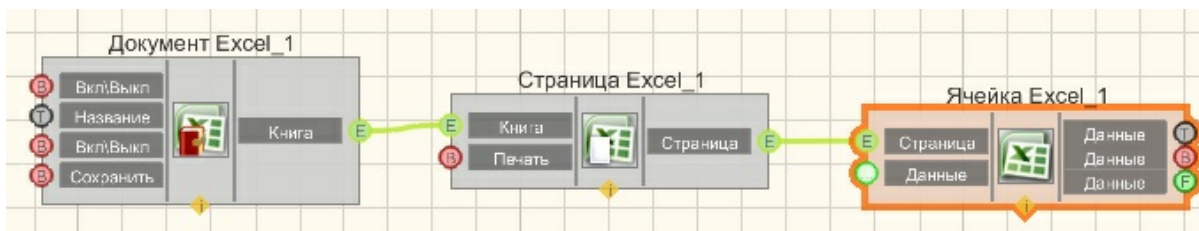
#### ➤ Частные свойства (в скобках значение, установленное по умолчанию):

- Line (1) - Номер строки.
- Row (1) - Номер столбца (указывается цифрой).

### Описание:

Данный компонент имеет некоторую особенность работы. Если, компонент получает одновременно 3 параметра: номер строки, номер столбца и значение которое необходимо записать, то независимо от количества изменений номера строки или номера столбца компонент не перейдет к нужной ячейке пока не получит значение на вход "Данные". То есть, если записать значение "3" в ячейку (1,2), а затем не меняя значение поменять номер ячейки, например (2,3), ничего не изменится. Но если, допустим подать на вход "Данные" значение 4, то компонент запишет в указанный excel-файл в ячейку (2,3), или в ту которая указана, значение 4.

Компонент "[Ячейка Excel](#)"<sup>[620]</sup> специфичен и является составным блоком, а не самостоятельным компонентом. Для записи данных в Excel-файл, необходимо использовать данный компонент совместно с компонентами "[Страница Excel](#)"<sup>[618]</sup> и "[Документ Excel](#)"<sup>[615]</sup>. Соединять компоненты нужно в следующей последовательности:



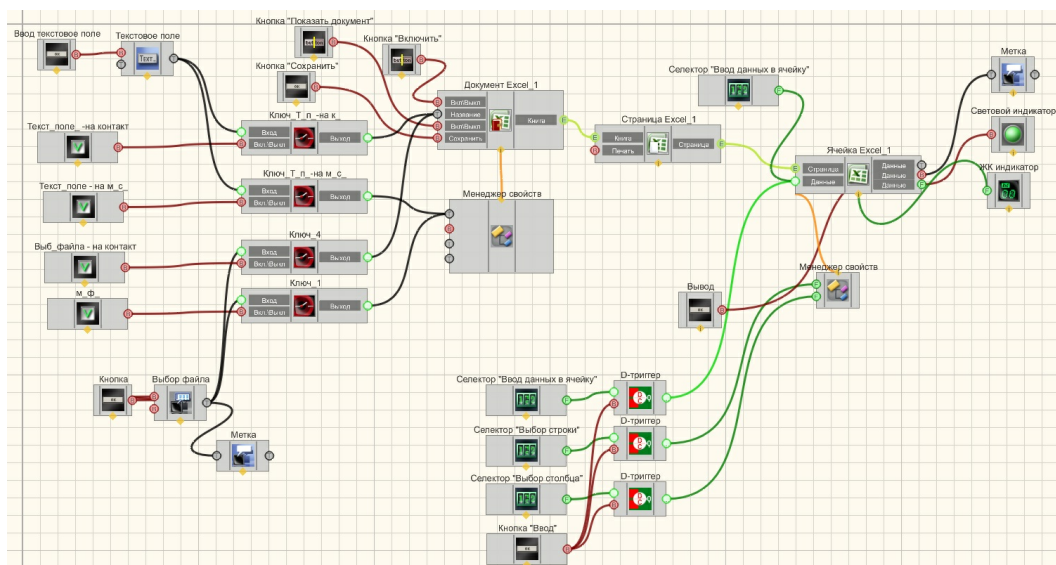
Несмотря на то, что компоненты "[Документ Excel](#)"<sup>[615]</sup>, "[Страница Excel](#)"<sup>[618]</sup> и "[Ячейка Excel](#)"<sup>[620]</sup> имеют входные и выходные контакты одного типа, с этих контактов поступают различающиеся данные. С компонента "[Документ Excel](#)"<sup>[615]</sup> поступает указатель на книгу, а с компонента "[Страница Excel](#)"<sup>[618]</sup> указатель на страницу. Поэтому нельзя соединить "[Документ Excel](#)"<sup>[615]</sup> и "[Ячейка Excel](#)"<sup>[620]</sup> напрямую.

Компонент стабильно работает с версией Microsoft Office Excel 2007.



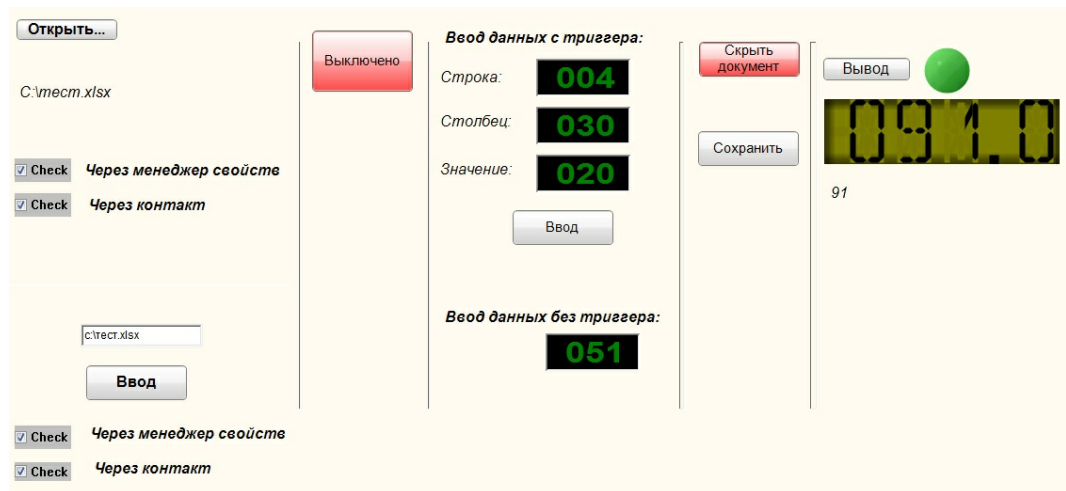
## Пример

### Проект в SCADA ZETView



### Результат работы проекта





Данный пример непосредственно в ZETView

Back Cover