

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«РАКУРС-ИНЖИНИРИНГ»







Заказчик: ООО «Ракурс-инжиниринг»

Договор №: В21.2018

Руководство пользователя/администратора

Программное обеспечение
анализа и диагностики вибрационного состояния
«Енисей-М»

САНКТ-ПЕТЕРБУРГ
2019

Наименование организации, предприятия	Должность исполнителя	Фамилия, инициалы	Подпись	Дата
ООО «Ракурс-инжиниринг»	Разработал	Лаушкин Д.С.		09.12.2019
	Проверил	Плетнев В.В.		09.12.2019
	Нормоконтроль	Ахмедов Н.Ш.		10.12.2019
	Утвердил	Ахмедов Н.Ш.		10.12.2019

Настоящее руководство устанавливает основные правила эксплуатации и назначение программного обеспечения, анализа и диагностики вибрационного состояния «Енисей-М»

Содержание

1	Структура ПО.....	6
1.1	Серверное ПО.....	6
1.2	База данных.....	7
1.3	Клиентское ПО.....	7
2	Серверное ПО.....	9
2.1	Сервер виброконтроля VibroSrv.....	9
2.1.1	Принцип работы.....	9
2.1.2	Настройка.....	9
2.2	Сервер интеграции данных FinsRouter.....	10
2.2.1	Принцип работы.....	10
2.2.2	Настройка.....	10
2.3	Сервер доступа к базе данных RawServer.....	10
2.3.1	Принцип работы.....	11
2.3.2	Настройка.....	11
3	Клиентское ПО VibroCtl.....	12
3.1	Настройка.....	12
3.2	Формат файлов мнемосхем.....	12
3.3	Клиентские скрипты.....	15
3.4	Структура базы данных конфигурации.....	17
4	Функции ПО VibroCtl.....	20
4.1	Основные элементы интерфейса.....	20
4.2	Элементы мнемосхем.....	21
4.3	Экран датчика.....	22
4.3.1	Вибрационный измерительный канал.....	22
4.3.2	Полюсной измерительный канал.....	23
4.4	Диаграммы.....	24
4.4.1	Меню диаграммы.....	25
4.4.2	Управление диаграммой мышью.....	26
4.4.3	Примеры отображения диаграмм.....	26
4.5	Экран «Анализ».....	27
4.6	Экран «Диагностика».....	28
4.7	Экран «События».....	29
4.8	Экран «Настройка».....	29
5	Жизненный цикл ПО.....	30
5.1	Требования к эксплуатационному персоналу.....	30
5.2	Устранение неисправностей.....	31
5.3	Техническая поддержка.....	32
5.4	Жизненный цикл ПО.....	33

Программное обеспечение «Енисей-М» (далее ПО) предназначено для архивации, отображения, обработки и анализа данных, поступающих от систем стационарного вибрационного контроля (ССВК). ПО является ядром автоматизированной системы мониторинга и диагностики вибрационного состояния гидроагрегатов. ССВК строится на базе универсальных измерительных контроллеров, объединённых в сеть Ethernet. Система может включать несколько контролируемых объектов, каждый из которых имеет один или несколько контроллеров.

В ПО реализован принцип клиент-сервер. Один или несколько серверов виброконтроля обеспечивают обмен данными с контроллерами ССВК и смежными системами, а также архивацию данных, дублируя друг друга. Клиенты виброконтроля взаимодействуют с основным в данный момент сервером виброконтроля, выполняя функции отображения текущих и архивных параметров ССВК, отображения аналитической и диагностической информации, конфигурирования системы.

1 СТРУКТУРА ПО

1.1 Серверное ПО

Серверное ПО состоит из сервера виброконтроля (сервера ввода-вывода данных), реализующего основные функции, сервера интеграции данных, сервера доступа к базе данных.

Сервер виброконтроля VibroSrv выполняет следующие функции:

- обмен данными с контроллерами ССВК;
- обмен данными со сторонними системами через сервер интеграции данных;
- архивация данных ССВК и сторонних систем;
- чтение спектрограмм и осциллограмм из контроллеров ССВК;
- предоставление текущих данных, спектрограмм и осциллограмм клиентскому ПО;
- диагностика связи с контроллерами;
- выполнение расчётных (в частности, диагностических) задач на основе подключаемых модулей (плагинов).

Сервер интеграции данных построен на базе ПО FinsRouter и обеспечивает двусторонний обмен текущими данными со сторонними системами по протоколам:

- Omron FINS over UDP/CLK/HostLink/Toolbus;
- Modbus TCP/RTU/ASCII;
- Siemens S7 ISO TCP;
- OPC DA 2.05;
- МЭК 60870-5-104;
- МЭК 61850;
- SNMP;
- другие, в том числе нестандартные, протоколы, с использованием дополнительных модулей (плагинов).

Сервер доступа к базе данных построен на базе ПО RawServer и обеспечивает доступ к архивным данным на чтение, а также доступ на чтение параметров конфигурации ССВК. Также сервер выполняет функции обслуживания базы данных. Основные функции сервера доступа к базе данных:

- многопользовательский доступ на чтение к архивным данным (измерениям, осциллограммам), накопленным сервером виброконтроля;
- доступ на чтение параметров конфигурации ССВК;
- оптимизация хранения архивных данных;
- синхронизация данных на дублированных серверах;
- сжатие и перенос на архивные сервера устаревших данных;
- удаление старых данных при недостатке дискового пространства.

Сервер VibroSrv является специфичным для ССВК, сервера FinsRouter и RawServer – стандартное ПО ГК «Ракурс» и могут быть использованы для «бесшовной» интеграции ССВК в АСУ ТП производства ГК «Ракурс». Документация на FinsRouter и RawServer приводится отдельными документами.

Всё серверное ПО представляет собой службы Microsoft Windows, запускается автоматически при старте Windows и не требует действий пользователя. Поддерживаются следующие версии Microsoft Windows в вариантах как 32bit, так и 64bit (не ниже редакции Professional, включая редакции Embedded для встраиваемых систем):

- Windows XP SP3, 7, 8, 8.1, 10;
- Windows Server 2003, 2008, 2012, 2016.

Серверное ПО состоит из следующих файлов:

- **VibroSrv.exe** – сервер виброконтроля;
- **VibroSrv.ini** – настройки сервера виброконтроля;
- **FinsSvc.exe** – сервер интеграции данных;
- **FinsSvc.ini** – настройки сервера интеграции данных;
- **RawSvc.exe** – сервер доступа к базе данных;
- **RawSvc.ini** – настройки сервера доступа к базе данных;
- **7z.dll** – библиотека сжатия данных (GNU LGPL, 7-Zip, www.7-zip.org);
- **sqlite3.dll** – библиотека поддержки SQLite (SQLite Development Team, www.sqlite.org).

1.2 База данных

База данных состоит из следующих частей:

- конфигурация ССВК;
- архив измерений (текущих данных);
- архив осциллограмм;
- история событий;
- мнемосхемы средств визуализации.

Конфигурация ССВК хранится в файлах БД SQLite и используется как клиентским, так и серверным ПО.

Архив текущих данных представляет собой двоичные файлы, расположенные в суточных каталогах. Данные ССВК за каждые сутки в UTC хранятся в отдельных каталогах. Измерения каждого контроллера хранятся в отдельном файле. История событий хранится в едином файле для всей системы. Доступ клиентского ПО к данным предусмотрен только на чтение.

Выполняется архивация полного набора текущих данных, что обеспечивает возможность восстановить полную вибрационную картину на произвольный момент времени. Для вибрационных параметров архивируются: 2А, СКЗ, амплитуды по всем диапазонам гармоник, слова состояния канала и вычисляемых параметров. Для полюсных параметров архивируются: минимум, максимум, значения по всем полюсам, слова состояния канала и значения. Архивируются также вычисляемые параметры и параметры, импортируемые из сторонних систем.

Данные за текущие сутки хранятся в исходном несжатом виде. Данные за предыдущие сутки могут автоматически сжиматься для экономии дискового пространства (задаётся настройкой RawServer). Сжатые данные распаковываются автоматически при доступе к ним и не требуют дополнительных действий клиентского ПО или пользователя.

Осциллограммы хранятся в архиве в отдельных двоичных файлах в суточных каталогах. Имя файла содержит имя контроллера, тип осциллограммы и время записи.

Мнемосхемы хранятся в виде отдельных текстовых файлов. Формат файлов мнемосхем описан далее.

1.3 Клиентское ПО

Клиентская часть ПО состоит из программы отображения данных (визуализации) ССВК **VibroCtl** и программы отображения архивных данных **Trends**.

Программа визуализации выполняет роль человеко-машинного интерфейса (HMI) ССВК. Основными её функциями являются:

- отображение в виде мнемосхем текущей вибрационной картины контролируемых объектов или отдельных их узлов;
- отображение сообщений и индикаторов предупредительной и аварийной сигнализации;
- отображение расчётных параметров и сообщений диагностических модулей;
- отображение и анализ вибрационных параметров (значения по диапазонам гармоник, спектрограммы, осциллограммы);
- отображение и анализ полюсных параметров (минимум, максимум, значения по полюсам, линейные и круговые диаграммы, осциллограммы);
- построение диаграмм и базовый многопараметрический анализ данных (воздушный зазор, магнитный поток, температура полюсов, траектория вала);
- диагностика связи с контроллерами и их работоспособности;
- настройка конфигурации ССВК.

Программа отображения архивных данных реализует следующие функции:

- отображение трендов измеренных и расчётных параметров;
- отображение истории событий с возможностью фильтрации;
- создание отчётных форм.

Программа Trends – стандартное ПО ГК «Ракурс», документация на него приводится отдельными документами. Программа Trends может использоваться как в составе ПО АРМ виброконтроля, так и отдельно, на любом компьютере, имеющем сетевой доступ к серверам ССВК.

На Рис.1.1 приводится общая структура ПО верхнего уровня ССВК, в простейшем случае содержащего один сервер и один АРМ. Допускается совмещение функций АРМ и сервера одним компьютером. Также возможны конфигурации, содержащие несколько АРМ и один или два сервера. В случае дублирования серверов, они работают параллельно и независимо друг от друга, при этом каждый АРМ может взаимодействовать только с одним сервером, который в данный момент является доступным. Роли основной-резервный между серверами в явном виде не распределяются.

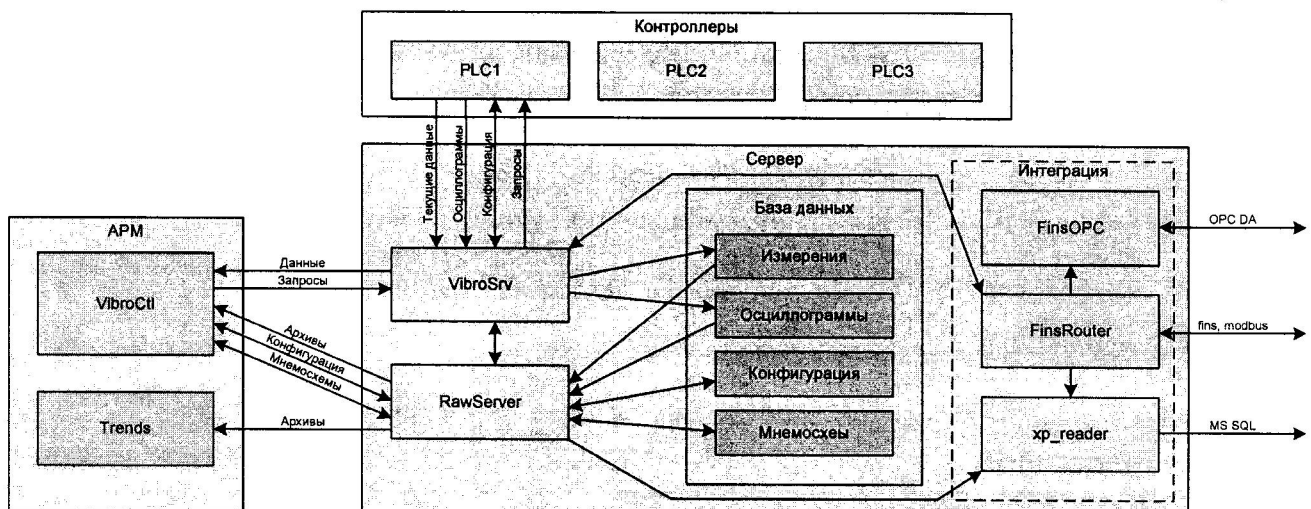


Рис. 1.1. Структура ПО.

2 СЕРВЕРНОЕ ПО

2.1 Сервер виброконтроля VibroSrv

Сервер виброконтроля является ключевым компонентом ПО, реализующим основной функционал и обеспечивающим взаимодействие всех программных компонентов между собой.

2.1.1 Принцип работы

Задачи сервера логически подразделяются на следующие группы:

- обмен данными с контроллерами;
- обмен данными со сторонними системами;
- обмен данными с клиентами;
- выполнение расчётных задач на основе плагинов;
- архивация текущих данных
- архивация событий;
- архивация осциллограмм;
- самодиагностика.

Взаимодействие со всеми контроллерами, сторонними системами, плагинами и клиентами реализовано в отдельных потоках для обеспечения равномерной вычислительной нагрузки и предотвращения взаимовлияния.

Непосредственно после запуска, сервер читает из базы данных конфигурацию ССВК, устанавливает соединения со всеми контроллерами, указанными в конфигурации, читает из них конфигурацию (общие настройки и состав сигналов), и начинает постоянный опрос текущих данных. С заданной периодичностью, наборы текущих данных записываются в архив. События, принимаемые от контроллера, записываются в архив без задержки, при этом время события присваивается контроллером.

При установлении соединения с контроллерами, а также при появлении признака обновления конфигурации контроллера, сервер обновляет базу данных конфигурации.

При наличии сторонних систем в качестве источников данных, производится подключение к ним напрямую или через FinsRouter и выполняется постоянное чтение описанных в конфигурации данных. Данные сторонних систем архивируются в отдельные таблицы базы данных с заданной периодичностью.

При подключении клиента (VibroCtl), сервер предоставляет ему запрашиваемую информацию. При запросе текущих данных по выбранному объекту, они возвращаются сразу же, так как обновляются постоянно и актуальны. При запросе спектрограмм и осциллограмм, они запрашиваются у контроллера и возвращаются клиенту.

При наличии плагинов в конфигурации сервера, после запуска сервера выполняется их инициализация, после чего они выполняются с заданной периодичностью. Плагины получают доступ к конфигурации ССВК и отдельных контроллеров, к текущим данным, спектрограммам и осциллограммам. Также плагины имеют возможность архивировать данные и обмениваться ими с клиентами и сторонними системами. Доступ к архивным данным реализуется через RawServer. Плагины представляют собой отдельные программные модули в виде динамической библиотеки (DLL).

2.1.2 Настройка

Настройки сервера хранятся в файле VibroSrv.ini.

[Settings]

```

Root=D:\SVK\
UseCache=1
SimMode=0
RequestPeriod=500
StoreLogFiles=90
Port=9800
    
```

- **Root** – путь к каталогу проекта, в котором находится ПО и следующие подкаталоги:
 - **Data** – база данных, в том числе конфигурация в файле Config.db;
 - **Cache** – копия данных из всех источников;
 - **Logs** – файлы системного журнала ПО;
 - **Pages** – мнемосхемы и изображения;
 - **Plugins** – серверные плагины.
- **UseCache** – сохранение и использование копии данных (0 – не использовать, 1 – использовать);
- **SimMode** – режим имитации с использованием сохранённых ранее данных (0 – нормальный режим, 1 – режим имитации);
- **RequestPeriod** – интервал запроса данных в миллисекундах;
- **StoreLogFiles** – время хранения файлов системного журнала в сутках.

2.2 Сервер интеграции данных FinsRouter

Сервер обеспечивает двунаправленный обмен текущими данными со сторонними системами и аккумуляцию данных из разнородных источников. Использование сервера опционально, он может быть использован для интеграции в том случае, если собственных средств ССВК по обмену со сторонними системами недостаточно.

2.2.1 Принцип работы

В конфигурации сервера настраиваются внешние подключения, используемые для импорта данных, а также внутренние области памяти (MemoryLink,mlink), используемые для аккумуляции данных. Также настраиваются области данных, постоянно читаемые из внешних подключений во внутренние области данных. Таким образом обеспечивается аккумуляция данных сторонних систем и поддержание их в актуальном состоянии.

Непосредственный доступ к внутренним областям данных предоставляется клиентам по протоколам Omron FINS или Modbus/TCP, поддержка других протоколов реализуется плагинами или отдельными серверными службами в составе FinsRouter.

Сервером поддерживается механизм плагинов, в том числе реализующих скриптовую обработку данных. С помощью механизма плагинов возможна реализация протоколов обмена данными, не входящих в список поддерживаемых, в том числе нестандартных.

2.2.2 Настройка

Настройка сервера детально описана в документации на FinsRouter.

2.3 Сервер доступа к базе данных RawServer

Сервер обеспечивает доступ к данным для компонентов ПО. Кроме того, сервер обеспечивает обслуживание базы данных.

2.3.1 Принцип работы

Сервер оперирует данными следующих типов:

- архивные данные в двоичном формате RAW (только чтение);
- конфигурация в формате SQLite;
- мнемосхемы в текстовом формате и изображения;
- шаблоны отчётов.

Архивные данные по запросу передаются в клиентские программы VibroCtl и Trends, а также могут быть переданы в плагины сервера виброконтроля и экспортированы в MS SQL Server через библиотеку xp_Reader.

Конфигурация используется всеми компонентами ПО. Клиентское ПО использует для доступа RawServer, сервер виброконтроля использует прямой доступ к конфигурации.

Мнемосхемы и изображения используются программой визуализации (VibroCtl).

Шаблоны отчётов используются Trends для построения отчётов.

2.3.2 Настройка

Настройки сервера хранятся в файле RawSvc.ini.

[DataBases]

```
Data=D:\SVK\Data
Config=D:\SVK\Data\Config.db
```

Расположение баз данных. В данном случае, имя Data используется для доступа к архивным данным, имя Config – для доступа к конфигурации. В файле VibroSrv.ini параметр Config должен указывать на базу данных конфигурации из данного файла.

[Ports]

```
SVK=9701
```

Перечень портов, по которым осуществляется доступ к данным. При этом, доступ к любой базе данных может быть осуществлён по любому порту. Должен быть указан как минимум один порт.

[Settings]

Общие настройки, в частности настройки ведения системного журнала.

```
LogFile=D:\SVK\Logs\RawSvc
StoreLogFiles=10
```

- **LogFile** – шаблон имени файла системного журнала (создаются ежедневные файлы добавлением даты к указанной в настройках строке);
- **StoreLogFiles** – время хранения файлов системного журнала в сутках.

Подробно настройки сервера описаны в документации на RawServer.

3 КЛИЕНТСКОЕ ПО VIBROCTL

3.1 Настройка

Настройки программы хранятся в файле `VibroCtl.ini`.

[Settings]

```
ProjectName=Красноярская ГЭС
Server=Server1
Port=9700
DataBase=9701
Plugins=\\Server1\KGES\Plugins
RequestPeriod=500
SensCodeWidth=84
Scale=0.8
Theme=DarkTheme
```

- **ProjectName** – имя проекта;
- **Server** – имя или IP-адрес сервера виброконтроля;
- **Port** – порт для подключения к серверу виброконтроля;
- **DataBase** – порт для подключения к RawServer;
- **Plugins** – расположение клиентских плагинов;
- **RequestPeriod** – интервал запроса текущих данных в миллисекундах;
- **SensCodeWidth** – ширина поля обозначения для датчиков (зависит от длины используемых в проекте обозначений);
- **Scale** – масштаб отображения мнемосхем (по умолчанию 1);
- **Theme** – цветовая тема приложения.

В секциях [Window] и [Sensor] запоминаются положение и размер окон программы и датчика.

3.2 Формат файлов мнемосхем

Файлы мнемосхем представляют собой текстовые файлы ASCII в национальной кодировке (1251 для русского языка), содержащие перечень объектов мнемосхем. Каждая строка файла содержит один объект и состоит из названия типа объекта и его параметров.

Параметры **X, Y** – координаты точки привязки объекта на мнемосхеме, как правило – левого верхнего угла объекта.

Параметры **Width, Height** – ширина и высота объекта.

Параметры ***Color**, указывающие цвет, задаются в виде шестнадцатеричного значения в формате **\$AARRGGBB**, где **AA** – прозрачность (00 - прозрачный, 80 – полупрозрачный, FF – непрозрачный), **RR** – красный, **GG** – зелёный, **BB** – синий компоненты цвета.

Параметр **Align** определяет выравнивание текста по горизонтали и вертикали. 0,3,6 – по левому краю; 1,4,7 – по середине текста; 2,5,8 – по правому; 0,1,2 – по верхнему краю; 3,4,5 – по центру; 6,7,8 – по нижнему краю.

0	1	2
3	4	5
6	7	8

Номера каналов (датчиков, аналоговых сигналов) указываются в текстовом формате **[Plant:]Plc.Index**, где **Plant** – номер агрегата, **Plc** – ID контроллера (нумеруются в пределах

одного агрегата, таким образом, мнемосхемы могут использоваться для нескольких однотипных агрегатов), **Index** – номер канала в контроллере (-1 – датчик частоты вращения). Для мнемосхем, относящихся к определённому агрегату, по умолчанию используется выбранный в данный момент агрегат.

Для дискретных сигналов адресация зависит от их источника. Для отображения в качестве дискретного сигнала признака **Status** аналогового сигнала из внешнего источника данных или из плагина используется формат для аналоговых сигналов. При этом значение может быть: 0 – нет данных (серый), 1 – информация (зелёный), 2 – предупреждение (жёлтый), 3 – авария (красный). Для отображения дискретных сигналов контроллера используется запись в формате #ID, где ID – идентификатор сигнала в таблице **Discr** конфигурации системы.

Предусмотрена возможность группировки объектов. Для этой цели служит объект **group**, описывающий группу, и дополнительный параметр «group=имя_группы», который может быть добавлен в качестве первого параметра перед списком параметров любого объекта. Для объектов внутри группы действуют относительные координаты (внутри прямоугольника группы) и используется агрегат по умолчанию, заданный в описании группы.

```
image (X, Y, FileName [ , FrameCount, [Channel, [Scale]]])
```

Изображение.

- **FileName** – имя файла изображения (поддерживаются форматы BMP, JPG, PNG);
- **FrameCount** – число кадров (если значение больше 1, изображение разбивается на указанное число кадров по горизонтали, из которых одновременно отображается только один);
- **Channel** – номер канала, значение которого определяет частоту смены кадров (может использоваться для анимации вращающихся частей);
- **Scale** – масштабный коэффициент для частоты смены кадров.

```
sensor (X, Y, Channel [ , Parts])
```

Датчик.

- **Channel** – номер канала;
- **Parts** – битовые флаги, определяющие отображение элементов датчика:
 - 1 – обозначение;
 - 2 – значение;
 - 4 – барграф;
 - 8 – предупредительная уставка;
 - 16 – аварийная уставка;
 - 32 – единицы измерения;
 - 64 – тип отображаемой величины (2А/СКЗ).

```
sensgroup (X, Y, Caption, List [ , Parts])
```

Группа датчиков.

- **Caption** – заголовок группы;
- **List** – список каналов в формате "Plc.Index", разделённых запятыми;
- **Parts** – битовые флаги, определяющие отображение элементов датчика.

```
link (X, Y [ , Flags [ , Discr ]])
```

Сноска для датчика или их группы, указывающая связь с другими элементами мнемосхемы. Применяется к предыдущему датчику в файле или к их группе.

- **X, Y** – точка привязки;
- **Flags** – флаги отображения (1 – рисовать точку);
- **Discr** – сноска для дискретного сигнала (1 – дискретный).

`panel (X, Y, Width, Height, PenColor, BrushColor, FontColor, FontSize, Align, Caption)`
или

`panel (X, Y, Width, Height, Style, Align, Caption)`

Панель с заголовком.

- **PenColor, BrushColor, FontColor** – цвета контура и фона панели и шрифта заголовка;
- **FontSize** – размер шрифта заголовка;
- **Align** – выравнивание заголовка внутри панели;
- **Caption** – текст заголовка;
- **Style** – название стиля из файла настроек.

`group (X, Y, Width, Height, Name, Plant)`

Группа объектов.

- **Name** – имя группы;
- **Plant** – номер контроллера, к которому будут относиться объекты группы.

`lamp (X, Y, Width, Height, Discr, Flags)`

Индикатор значения дискретного сигнала.

- **Discr** – номер дискретного сигнала;
- **Flags** – флаг отображения (0 – индикатор в виде кружка на прямоугольнике с названием сигнала, 1 – индикатором является сам фон прямоугольника).

`lampgroup (X, Y, Width, Height, Caption, List, Flags)`

Группа индикаторов значений дискретных сигналов.

- **Caption** – заголовок группы;
- **List** – список номеров дискретных сигналов;
- **Flags** – флаг отображения.

`libitem (X, Y, Lib, Item, Params)`

Плагин в виде динамической библиотеки (см. «Клиентские плагины»).

- **Lib** – имя файла динамической библиотеки (должен находиться в папке плагинов, указанной в настройках);
- **Item** – имя процедуры в библиотеке, выполняющей отрисовку изображения, таким образом в одной библиотеке может находиться несколько разных объектов;
- **Params** – перечень параметров, разделённых запятыми, передающийся процедуре отрисовки.

`script (X, Y, Width, Height, Source, Params)`

Плагин в виде скрипта (см. «Клиентские скрипты»).

- **Source** – файл скрипта;
- **Params** – перечень параметров, разделённых запятыми, передающийся скрипту.

3.3 Клиентские скрипты

Основное назначение – предоставить пользователю возможность развития функционала системы в части отображения, без использования дополнительных средств разработки. Клиентский скрипт представляет собой файл с кодом на одном из языков программирования, в котором реализована отрисовка графического объекта и его реакция на внешние события. Для скриптов реализованы механизмы доступа к данным системы и механизмы отображения, основанные на GDI+.

Поддерживаются языки Pascal, C, Java, Basic с учётом некоторых ограничений синтаксиса. В качестве интерпретатора используется FastScript компании FastReports.

Файл скрипта состоит из обязательной секции инициализации и может содержать процедуры с предопределёнными именами, отвечающие за отрисовку и за реакцию на внешние события от мыши.

Общая структура скрипта в синтаксисе Pascal:

```

var // Глобальные переменные

procedure Paint;
begin
    // Отрисовка
end;

procedure MouseMove(Shift,X,Y: Integer);
begin
    // Реакция на перемещение мыши
end;

procedure MouseDown(Keys,X,Y: Integer);
begin
    // Реакция на нажатие клавиши мыши
end;

procedure MouseUp(Keys,X,Y: Integer);
begin
    // Реакция на отпускание клавиши мыши
end;

procedure MouseWheel(Keys,X,Y,Delta: Integer);
begin
    // Реакция на колесо мыши
end;

begin
    // Секция инициализации, выполняется при создании объекта
end.
```

Объект создаётся при открытии страницы, на которой он расположен, и уничтожается при переходе на другую страницу.

Стандартные функции языков программирования дополнены следующими константами, переменными и процедурами.

```

Draw: Integer = 1
Fill: Integer = 2
Close: Integer = 4
```

Флаги отрисовки, определяющие будет ли выполняться рисование контура фигуры, заливка фигуры и замыкание контура фигуры. Могут суммироваться.

`ActivePlant: Integer`

Номер текущего выбранного агрегата.

`procedure Log(Msg: String)`

Вывод строки в лог приложения.

`procedure AntiAliasing(State: Boolean)`

Включение или отключение сглаживания при отрисовке.

`procedure Poly(X: array of Single; Flags: Integer = Draw)`

Рисование полилинии или многоугольника, где **X** представляет собой массив координат точек в формате [X0,Y0, X1,Y1 ... Xn,Yn].

`procedure Curve(X: array of Single; Tension: Single = 1; Flags: Integer = Draw)`

Рисование гладкой кривой, где **Tension** – условная жёсткость кривой.

`procedure Rect(X,Y,Width,Height: Single; Flags: Integer = Draw)`

Рисование прямоугольника.

`procedure Ellipse(X,Y,Width,Height: Single; Flags: Integer = Draw)`

Рисование эллипса или окружности.

`procedure Arc(X,Y,Width,Height,Start,Sweep: Single)`

Рисование дуги эллипса, где **Start** и **Sweep** – начальный угол и угловой размер дуги в градусах.

`procedure Pie(X,Y,Width,Height,Start,Sweep: Single; Flags: Integer = Draw)`

Рисование сектора эллипса.

`procedure TextXY(X,Y: Single; Str: String; Align: Integer = 0)`

Вывод текста в определённой точке с выравниванием **Align** (см. «Формат файлов мнемосхем»). Текст выводится с использованием текущей кисти.

`procedure TextRect(X,Y,Width,Height: Single; Str: String; Align: Integer = 0)`

Вывод текста в определённом прямоугольнике с отсечкой по его границам.

`procedure Brush(Color: Integer)`

Задание цвета кисти для заливки фигур и вывода текста.

`procedure Pen(Color: Integer; Width: Single = 1)`

Задание цвета и толщины пера для отрисовки контуров фигур.

`procedure Font(Typeface: String; Size: Single; Style: Integer = 0)`

Задание типа, размера и стиля шрифта для вывода текста.

`function GetSensValue(Plant,Plc,Index,Pos: Integer): Integer`

Получение определённого параметра из массива данных датчика. Возвращается элемент массива **Pos** из датчика номер **Index** контроллера **Plc** агрегата **Plant**.

```
procedure GetSensData(Plant,Plc,Index: Integer; var Data: array of Integer)
```

Получение массива данных датчика (100 целочисленных значений определённой структуры для вибрационных и полюсных каналов).

Вибрационные: [2] – частота в мГц, [3] – 2А, [5] – СКЗ.

Полюсные: [2] – частота в мГц, [3] – Min, [4] – Max, [6..95] – значения по полюсам.

```
procedure GetOsc(Plant,Plc,Index: Integer; var Data: array of Integer)
```

Запрос осциллограммы. Операция выполняется асинхронно и актуальные данные фактически будут получены только при следующем вызове функции отрисовки.

3.4 Структура базы данных конфигурации

База данных конфигурации хранится в формате SQLite на сервере и состоит из следующих основных таблиц:

- **Plants** – список контролируемых объектов ССБК (гидроагрегатов);
- **Plcs** – список контроллеров всех объектов, а также внешних подключений и плагинов;
- **Channels** – список датчиков (измерительных каналов) всех объектов;
- **ChannelTypes** – список типов измерительных каналов;
- **Discr** – список дискретных сигналов;
- **Pages** – список мнемосхем;
- **DiagChannels** – список диагностических параметров;
- **Consts** – список констант для задач диагностики.

```
Plants (id, Name, Poles, Slots, ImpBlades, GuideBlades, Rotation, OscPerDay, GateSrv)
```

- **id** – ID агрегата (0..255);
- **Name** – наименование агрегата;
- **Poles** – число полюсов ротора;
- **Slots** – число пазов статора;
- **ImpBlades** – число лопастей рабочего колеса;
- **GuideBlades** – число лопаток направляющего аппарата;
- **Rotation** – направление вращения (0 – по часовой стрелке, 1 – против);
- **OscPerDay** – количество осциллограмм, автоматически сохраняемых в день (сохраняются через равные промежутки времени, начиная с полуночи);
- **GateSrv** – адрес промежуточного сервера, через который подключены контроллеры.

```
Plcs (Plant, id, Active, Type, Name, Addr, Version)
```

- **Plant** – ID гидроагрегата;
- **id** – ID контроллера в пределах агрегата (0..255);
- **Active** – признак активации (0 – игнорируется, 1 – активен);
- **Type** – тип контроллера или другого источника данных (1 – контроллер NI, 2 – внешнее подключение, 3 – плагин, 4 – AS-02);
- **Name** – наименование контроллера;

- **Addr** – адрес источника данных, зависит от его типа (1,4 – IP-адрес контроллера, 2 – строка подключения, 3 – имя библиотеки DLL с указанием пути относительно папки Root);
- **Version** – версия конфигурации.

Channels (Plant, Plc, Index, Code, Name, Units, Scale, Angle, Signal, Min, Max, Config)

- **Plant** – ID агрегата;
- **Plc** – ID контроллера;
- **Index** – номер в контроллере (0..255);
- **Code** – краткое обозначение;
- **Name** – полное наименование;
- **Units** – единицы измерения;
- **Scale** – масштаб для пересчёта из целочисленного значения в контроллере;
- **Angle** – угловое расположение датчика в градусах;
- **Signal** – единицы измерения входного сигнала (для Plc тип 1);
- **Min** – минимум шкалы;
- **Max** – максимум шкалы;
- **Config** – параметры канала, зависит от типа контроллера (1 – шестнадцатеричное представление блока настроек канала в контроллере; 2 – адрес параметра в формате адрес/тип#номер, где адрес/тип – начальный адрес и тип блока данных в контроллере (например, D100/SH, W200/F, DB10:20/FS и т.д.), номер – номер элемента данных в блоке, нумерация с 0; 3 – не используется; 4 – тип канала: V – вибрационный, P – полюсной).

Discr (id, Code, Name, Plc, Addr, Color)

- **id** – ID сигнала;
- **Code** – краткое обозначение;
- **Name** – полное наименование;
- **Plc** – ID контроллера (каждый сигнал может быть привязан ко всем контроллерам);
- **Addr** – битовый адрес сигнала в блоке данных контроллера в формате **слово.бит**;
- **Color** – цвет отображения сигнала (1 – зелёный, 2 – жёлтый, 3 – красный).

Pages (Plant, id, Type, Name, Page)

- **Plant** – ID объекта (-1 – для всех агрегатов, 0 – общестанционная);
- **id** – ID мнемосхемы (уникальный в пределах агрегата);
- **Type** – в данной версии не используется, должно быть 1;
- **Name** – наименование мнемосхемы для отображения на закладке;
- **Page** – имя файла мнемосхемы (без расширения).

DiagChannels (id, Plc, Index, Code, Name, ShortName, Vibr, Units, Min, Max, Scale, Angle, Sign)

- **id** – уникальный ID параметра;
- **Plc** – ID контроллера в составе гидроагрегата;
- **Index** – ID сигнала в контроллере;
- **Code** – условное обозначение, позволяющее идентифицировать сигнал;
- **Name** – полное наименование;
- **ShortName** – краткое наименование;
- **Vibr** – признак вибрационного параметра (1 – вибрационный, 0 – нет);

- **Units** – единицы измерения;
- **Min** – минимум шкалы;
- **Max** – максимум шкалы;
- **Scale** – масштаб для пересчёта из целочисленного значения в контроллере;
- **Angle** – угловое расположение датчика в градусах;
- **Sign** – не используется.

Сигналы, перечисленные в данной таблице, используются для диагностики. По ним определяется режим работы гидроагрегата и выполняется архивация среднеминутных значений для последующего анализа (экран Анализ) и для использования диагностическими плагинами. Предполагается однотипное расположение сигналов для всех гидроагрегатов.

Поле **Code** используется для идентификации сигналов, независимо от конфигурации сигналов в контроллере. Сервером виброконтроля используются следующие обозначения:

- **P** – активная мощность, МВт;
- **H** – напор, м;
- **F** – частота электрическая или пересчитанная в неё частота вращения, Гц;
- **U** – напряжение статора, кВ;
- **NA** – открытие направляющего аппарата, %.

Также обозначение может использоваться диагностическими плагинами для связи необходимого диагностического параметра с определённым сигналом в контроллере.

Поле **Vibr** указывает тип параметра; для вибрационных, дополнительно к интегральной оценке вибрации (2A, СКЗ), сохраняются параметры до 30 основных частотных составляющих.

Поля **ShortName**, **Units** используются для отображения.

Поля **Min**, **Max**, **Scale**, **Angle** справочные и могут быть использованы диагностическими плагинами.

`Consts (Name, Units, Values, Comment, ReadOnly, AlgNum, id)`

- **Name** – условное обозначение константы;
- **Units** – единицы измерения;
- **Values** – перечень значений для каждого ГА в формате «ГА=значение» через запятую; если указано только одно значение – оно используется для всех гидроагрегатов;
- **Comment** – описание константы;
- **ReadOnly** – признак возможности изменения пользователем (1 – изменение запрещено);
- **AlgNum** – номер или номера алгоритмов (используется плагинами);
- **Id** – ID константы (используется плагинами).

4 ФУНКЦИИ ПО VIBROCTL

Программа визуализации выполняет роль человеко-машинного интерфейса ССВК и предназначена для получения, математической обработки, отображения и документирования данных виброконтроля. Источником данных для программы является сервер виброконтроля (текущие измерения, сообщения, спектрограммы и осциллограммы, диагностические данные), а также сервер доступа к базе данных (архивные данные).

4.1 Основные элементы интерфейса

Управление программой осуществляется через привычный для пользователей ОС Windows графический интерфейс. Вид главного окна VibroCtl приведен на Рис. 4.1.

В верхней части окна программы находятся:

- главное меню;
- меню выбора гидроагрегата;
- меню выбора мнемосхемы;
- мнемосхема или другой информационный экран.

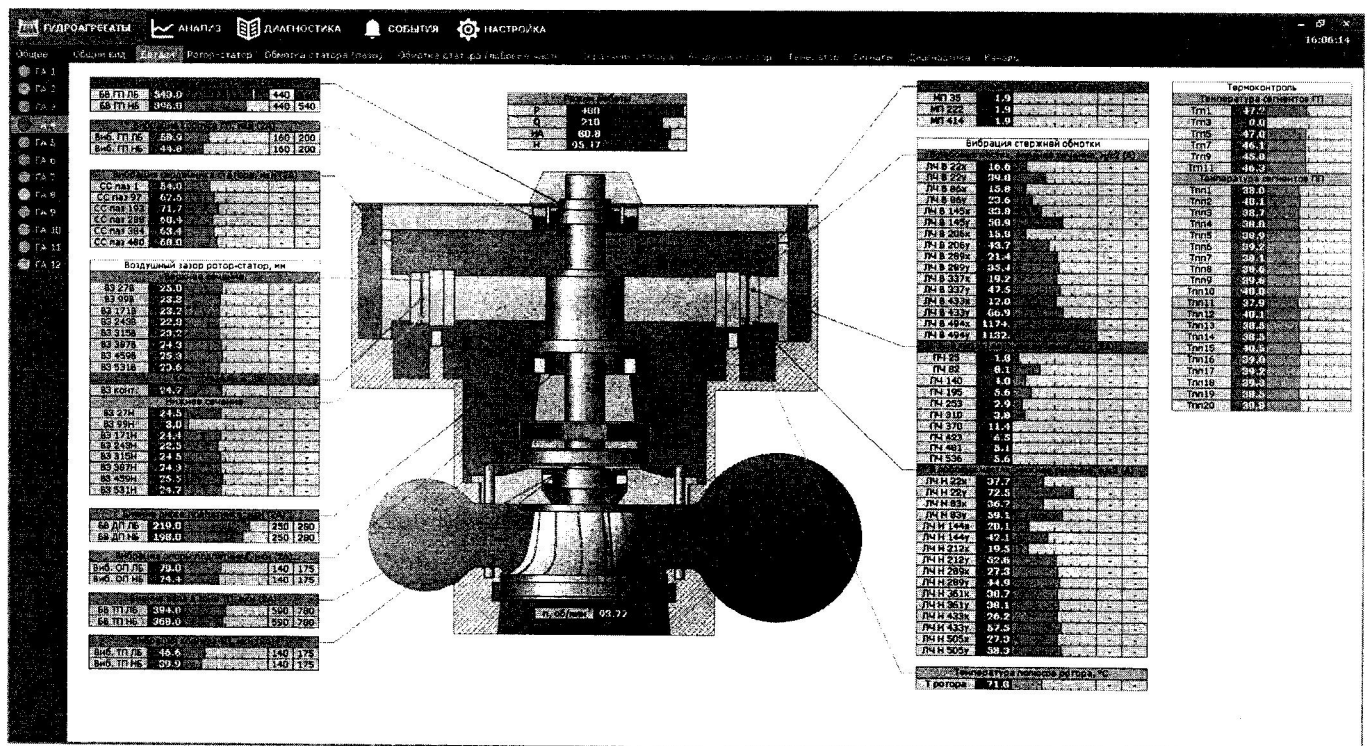


Рис. 4.1. VibroCtl. Общий вид.

Главное меню позволяет переключаться между различными информационными экранами.

Гидроагрегаты: экран отображает мнемосхемы контролируемых объектов. Мнемосхемы могут содержать несколько гидроагрегатов, один выбранный в меню гидроагрегат или отдельные его узлы. На мнемосхемах отображаются текущие данные, получаемые от контроллеров ССВК, внешних источников и плагинов.

Анализ: экран предоставляет доступ к архивным осциллограммам.

Диагностика: экран предоставляет возможность определить время работы гидроагрегата в различных режимах по мощности и напору и оценить изменение во времени вибрационных параметров при работе в сходных режимах.

Настройка: экран предоставляет данные самодиагностики сервера виброконтроля и пользовательские настройки.

Меню выбора агрегата позволяет переключаться между гидроагрегатами, а также выбирать общестанционные экраны. Обновление данных на экране пользователя происходит сразу же после выбора интересующего объекта.

Меню выбора мнемосхемы позволяет переключаться между различными мнемосхемами выбранного гидроагрегата или общестанционными мнемосхемами. Мнемосхема «Каналы» отображает полный перечень измерительных каналов выбранного гидроагрегата, включая данные из внешних источников.

4.2 Элементы мнемосхем

Элементы мнемосхем отображают текущую вибрационную картину гидроагрегата, как в цифровом, так и в графическом виде.

Основными динамическими объектами мнемосхем являются:

- датчики и их группы;
- дискретные сигналы и их группы;
- сноски;
- анимированные изображения;
- пользовательские объекты.




Изображение **датчика** состоит из следующих элементов (Рис. 4.2.1):

- **обозначение** – краткое название измерительного канала;
- **значение** – численное значение измеряемого параметра, зависящее от настроек (для вибрационных параметров может быть выбрано отображение значений 2А или СКЗ, а для полюсных – минимальное или максимальное значение по полюсам). Красный фон поля значения говорит о неисправности измерительного канала;
- **барграф** с уставками – графически отображает значение в диапазоне от минимального до максимального, цветом отображается достижение предупредительного или аварийного уровня сигнала;
- **уставки** – численные значения предупредительной и аварийной уставки (или прочерк, если уставки не заданы);
- **единицы измерения;**
- **тип отображаемой величины** – указывается для вибрационных каналов (2А или СКЗ) и зависит от настроек измерительного канала в контроллере.



Рис. 4.2.1. Датчик.

Цвет барграфа показывает состояние сигнализации:

-  серый – нет данных;
-  зелёный – нет предупредительных и аварийных сигналов;
-  жёлтый – предупредительный сигнал;

- красный – аварийный сигнал.

Группа датчиков представляет собой набор датчиков с общим заголовком (Рис. 4.2.2). В группу объединяются датчики, измеряющие один и тот же физический параметр одного узла гидроагрегата и имеющие одинаковые пределы измерения. Цвет заголовка группы датчиков отображает наличие предупредительных или аварийных сигналов по всем датчикам группы.

БВ ГП ЛБ	543.0	440	540
БВ ГП НБ	575.0	440	
Виб. ГП ЛБ	110.9	160	200
Виб. ГП НБ	82.2	160	200

Рис. 4.2.2. Группа датчиков.

Дискретные сигналы и их группы отображают текущее состояние дискретных сигналов или иных битов признаков. Отсутствие сигнала отображается серым, наличие сигнала отображается цветом, заданным для него в конфигурации. Дискретные сигналы могут отображаться как представлено на Рис. 4.2.3 или прямоугольником, меняющим цвет фона указанным образом.

Авария	●
Предупреждение	○
Неисправность	●
Штатный останов	●
Аварийный останов	●

Рис. 4.2.3. Группа дискретных сигналов.

Сноска представляет собой линию, соединяющую датчик или группу датчиков с другими объектами на мнемосхеме, обеспечивая визуальную привязку датчиков к контролируемым элементам. Связь меняет цвет в зависимости от состояния сигнализации привязываемых датчиков.

Анимированные изображения позволяют наглядно отобразить движение, например, вращение рабочего колеса. Видимость движения достигается сменой кадров, частота смены кадров связана с параметром, определяющим фактическую скорость движения, таким образом, скорость движения на экране соответствует фактической скорости движения (вращения).

4.3 Экран датчика

При нажатии на изображение вибрационного или полюсного датчика на мнемосхеме, будет открыт экран с детальной информацией по нему и текущей осциллограммой сигнала. Состав информации и вид осциллограммы зависят от типа измерительного канала (вибрационный или полюсной) и от типа источника данных (контроллер NI или AS-02).

4.3.1 Вибрационный измерительный канал

К вибрационным относятся следующие типы каналов:

- абсолютные виброперемещения (вибрация);
- относительные виброперемещения (биение).

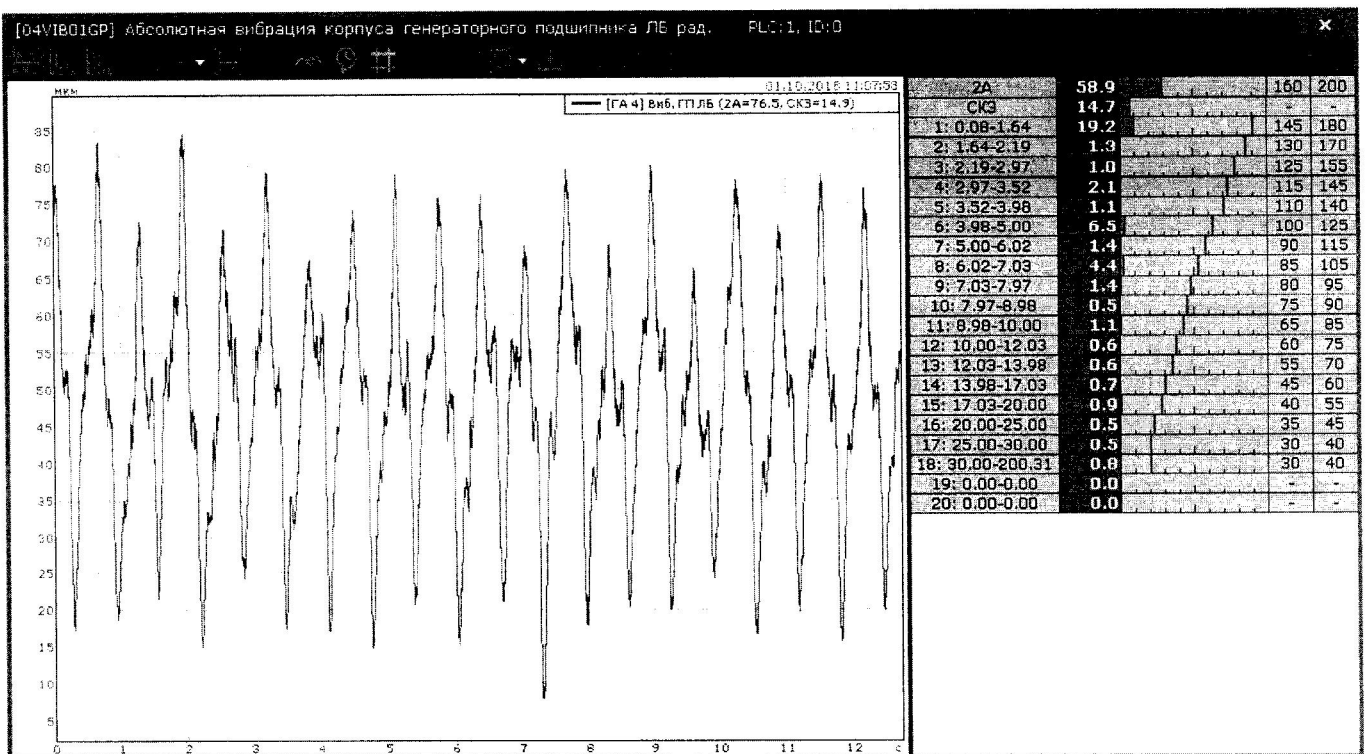


Рис. 4.3.1. Экран вибрационного канала измерения.

В левой части экрана расположена осциллограмма сигнала.






В правой части экрана кнопкой  может быть открыт список контролируемых параметров сигнала: интегральные значения 2А и СКЗ, 20 настраиваемых диапазонов гармоник. По каждому из параметров отображаются заданные уставки и признаки их срабатывания.

Диаграмма может отображать следующие виды информации:

-  осциллограмма;
-  спектрограмма;
-  спектральный анализ с выделением ключевых частот.
-  орбита

4.3.2 Полюсной измерительный канал

К полюсным каналам измерения относятся следующие каналы:

- воздушный зазор статор-ротор;
- магнитный поток;
- температура полюсов ротора.

При нажатии на датчик полюсного параметра, будет открыт экран с детальной информацией, в целом аналогичный экрану вибрационного параметра. В левой части экрана расположена осциллограмма. В правой – список значений по полюсам, а также их минимальное и максимальное значения. Уставки отображаются по минимальному и максимальному значениям, по отдельным полюсам уставки не предусмотрены.

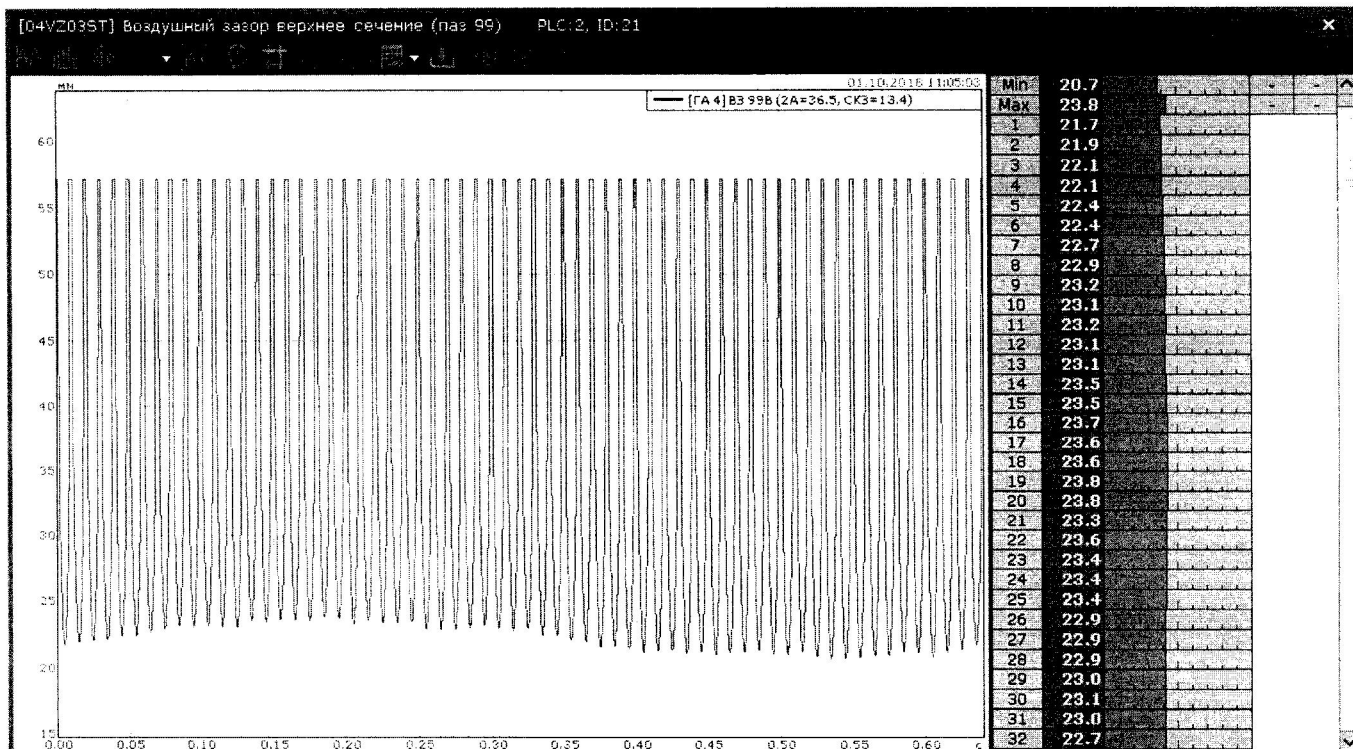





Рис. 4.3.2. Экран полюсного канала измерения.

Диаграмма может отображать следующие виды информации:

-  осциллограмма за один оборот;
-  полюсная диаграмма;
-  круговая диаграмма.

4.4 Диаграммы

Диаграммы предназначены для графического представления измеряемых значений сигналов. В зависимости от типа сигнала, возможно различное его представление на диаграмме.

Вибрационные каналы:

- осциллограмма – график изменения сигнала во времени;
- орбита – график изменения одного из пары ортогональных сигналов, относительно другого (для биения – траектория движения контролируемого узла);
- спектрограмма – спектральный состав осциллограммы сигнала;
- спектральный анализ – спектрограмма с выделением ключевых частот.

Полюсные каналы:

- осциллограмма – график изменения сигнала во времени за один оборот;
- полюсная диаграмма – диаграмма значений сигнала по полюсам;
- круговая диаграмма – осциллограмма в полярных координатах.

Функционал диаграмм зависит от того, отображается ли текущая осциллограмма (экран датчика) или архивная осциллограмма (экран анализа). Текущая осциллограмма в экране датчика автоматически обновляется при обновлении в контроллере (обновление может быть отключено).

Управление диаграммой осуществляется как в пиктографическом меню, так и мышью в сочетании с клавишами Ctrl и Shift.

4.4.1 Меню диаграммы

Общий вид меню представлен на рисунках выше. Состав меню зависит от доступного функционала для данного типа канала и данного экрана.



Рисунок 4.4.1 – Пример общего вида меню диаграммы.

	Осциллограмма – график изменения сигнала во времени.
	Спектрограмма – спектральный состав осциллограммы сигнала.
	Спектральный анализ – спектрограмма с выделением ключевых частот.
	Орбита – график изменения одного из пары ортогональных сигналов, относительно другого, для вибрационных параметров
	Круговая диаграмма – осциллограмма полюсного сигнала в полярных координатах.
	Выбор вида диаграммы – доступность видов диаграммы зависит от её типа.
	Автоматическое масштабирование – отображение полного размаха сигнала по осям X и Y, отключается при операциях ручного масштабирования.
	Отметчики оборотов на диаграмме – отображение красными точками моментов прохождения отметчика оборотов.
	Абсолютная шкала времени – перевод шкалы времени из относительной с начала осциллограммы в абсолютное местное время.
	Раздельные оси для разных сигналов – отображение каждого из сигналов на отдельной оси Y с возможностью индивидуального масштабирования.
	Курсор-рамка – отображение перемещаемых пользователем вспомогательных линий, параллельных осям координат.
	Гармонический курсор – отображение на спектрограмме перемещаемых пользователем линий для определения кратности частот.
	Характеристические частоты – отображение на спектрограмме линий характеристических частот данного гидроагрегата.
	Выбор числа диаграмм на экране – для экрана Анализ выбор числа и расположения диаграмм на экране.
	Табличное представление данных – отображение данных выбранного сигнала в табличном виде.
	Сохранение данных и изображения – сохранение графического изображения диаграммы (формат PNG) и её исходных данных (формат CSV) в файлы в заданной папке (без запроса у пользователя имени файла).
	Включение/отключение обновления данных – позволяет блокировать обновление данных для экрана датчика.
	Сохранение осциллограммы – отправляет команду серверу на сохранение в базе данных текущей осциллограммы с пометкой «пользовательская».



Отображение дополнительных параметров измерительного канала.

4.4.2 Управление диаграммой мышью

Перемещение – правой кнопкой мыши переместить за свободное поле диаграммы.

Перемещение по осям – левой кнопкой мыши переместить выбранную ось или колесо мыши с клавишей Shift (ось X) или без неё (ось Y).

Масштабирование по осям – удерживая Ctrl, левой кнопкой мыши переместить выбранную ось или колесо мыши с клавишей Shift (ось X) или без неё (ось Y).

Увеличение фрагмента – выделить левой кнопкой мыши нужный фрагмент по направлению из левого верхнего угла в правый нижний.

Просмотр значения – подвести курсор мыши к точке на диаграмме.

Установка сноски (метки значения) – выделить левой кнопкой мыши точку на диаграмме (сноски удаляются при обновлении диаграммы, в частности, при автоматическом обновлении в экране датчика).

Перенос сноски – перенести сноску левой кнопкой мыши.

Удаление сноски – щёлкнуть колесом мыши по сноске.

Выбор частоты на спектрограмме – удерживая Ctrl, указать точку на спектрограмме (значение частоты появится в левом верхнем углу поля диаграммы, в дальнейшем оно может быть использовано для частотной фильтрации).

Отменить выбор частоты – щёлкнуть колесом мыши по значению выбранной частоты.

Специфическое управление для экрана Анализ:

Выбор диаграммы – любой щелчок мыши в поле диаграммы (легенда выбранной диаграммы отображается жёлтым, операции в меню относятся к выбранной диаграмме).

Удаление параметра – щёлкнуть колесом мыши по параметру в легенде.

Удаление всех параметров – удерживая Ctrl, щёлкнуть колесом мыши по любому параметру в легенде.

Расширение поля диаграммы на весь экран – двойной щелчок по полю диаграммы.

Перенос диаграммы – левой кнопкой мыши за легенду перенести диаграмму в свободную ячейку.

Клонирование диаграммы – удерживая Ctrl, левой кнопкой мыши за легенду перенести диаграмму в свободную ячейку (данные в клонированных диаграммах будут обновляться синхронно).

«Заморозка» параметра – удерживая Ctrl, указать левой кнопкой мыши нужный параметр в легенде (замороженный параметр выделяется голубым и перестаёт перемещаться по оси времени).

4.4.3 Примеры отображения диаграмм

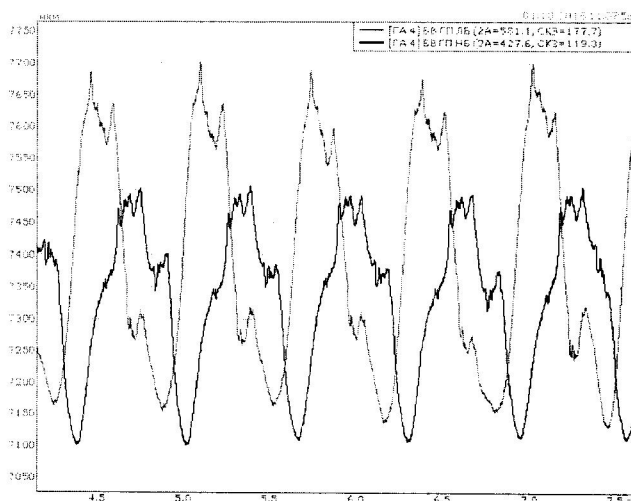


Рис. 4.4.2. Несколько параметров на диаграмме.

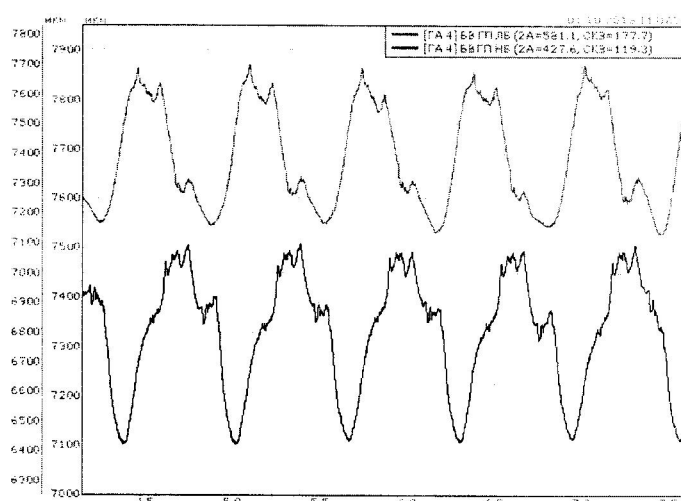


Рис. 4.4.3. Параметры на отдельных осях.

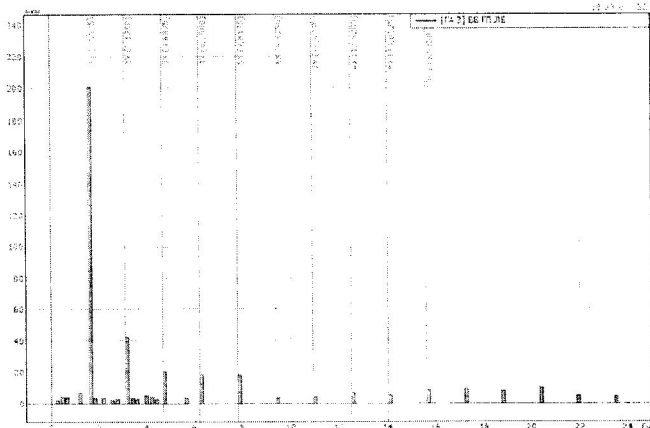


Рис. 4.4.4. Спектральный анализ и гармонический курсор.

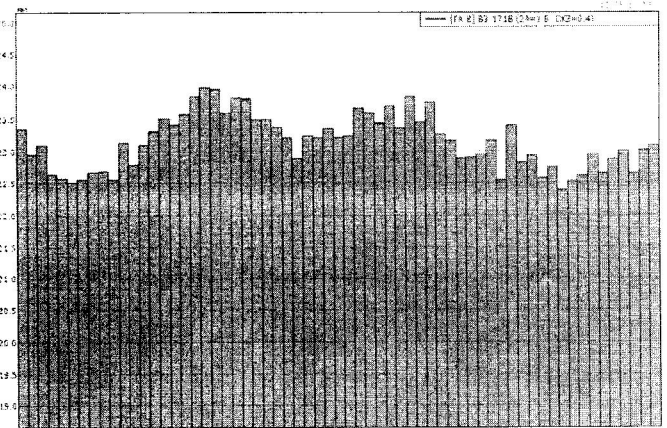


Рис. 4.4.5. Полюсная диаграмма.

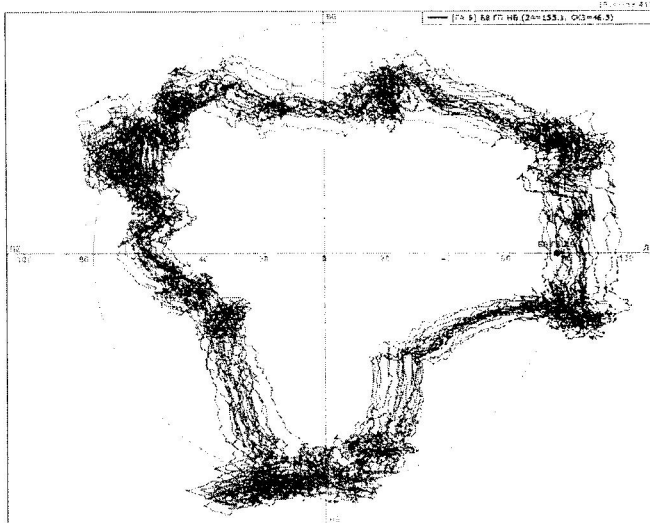


Рис. 4.4.6. Орбита.

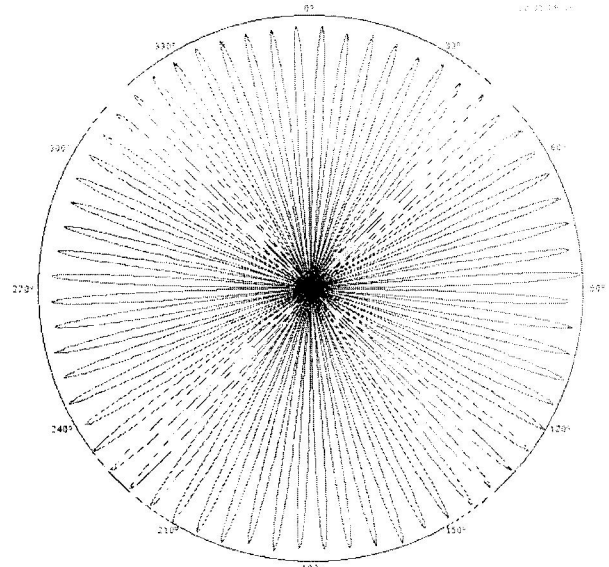


Рис. 4.4.7. Круговая диаграмма.

4.5 Экран «Анализ»

Экран предоставляет доступ к архивным осциллограммам, накопленным в базе данных. Осциллограммы по принципам сохранения могут быть трёх типов:

- **периодическая** – автоматически сохраняется через равные настраиваемые промежутки времени (только при вращении гидроагрегата);
- **пользовательская** – сохраняется по команде пользователя;
- **аварийная** – автоматически сохраняется при наступлении аварийного события и формируется объединением в одну непрерывную осциллограмму нескольких осциллограмм, записанных как до, так и после наступления события.

Левая часть экрана предназначена для выбора осциллограмм. Она содержит **дерево сигналов**, упорядоченных по типу, **календарь** для выбора отрезка времени, **список осциллограмм** за данный отрезок времени.

Правая часть представляет собой **рабочую область**, разделённую на ячейки для диаграмм. При этом каждая диаграмма может содержать данные по нескольким параметрам, относящимся к разным гидроагрегатам. Совмещение на одной диаграмме вибрационных и полюсных параметров не предусмотрено.

Для анализа осциллограммы необходимо выбрать нужный сигнал в дереве сигналов и перетащить его мышью в свободную ячейку в рабочей области. Можно перетащить всю группу сигналов. Затем необходимо в календаре выбрать отрезок времени, за который предполагается просматривать осциллограммы. Месяцы в календаре можно менять как стрелками, так и колесом мыши. После выбора отрезка времени, ниже будет отображён список осциллограмм, сохранённых за указанные даты. Для пользовательских и аварийных осциллограмм указывается номер гидроагрегата, для которого они сохранены (в скобках), и код записи (2 – пользовательская, 3 – аварийная). Выбранная в списке осциллограмма будет отображена в выбранной диаграмме.

В случае одновременной работы с несколькими диаграммами, для использования меню или для выбора отображаемой осциллограммы необходимо сначала выбрать диаграмму, к которой эти действия будут применены – для этого достаточно любого нажатия кнопки мыши в поле диаграммы.

Раскладка экрана на ячейки с диаграммами может быть изменена в любой момент соответствующей кнопкой в меню. Открытые диаграммы при этом будут сохранены. Временное увеличение любой диаграммы на весь экран возможно двойным щелчком по ней, аналогично и возврат обратно. Разделители ячеек диаграмм можно перетаскивать, меняя размеры диаграмм. При изменении размера окна программы и при изменении раскладки экрана, размеры диаграмм будут автоматически сделаны одинаковыми.

4.6 Экран «Диагностика»

Экран позволяет проанализировать изменение вибрационных параметров на длительных промежутках времени в сопоставимых режимах работы гидроагрегата. Таким образом можно визуально оценить как степень деградации вибрационных характеристик в ходе эксплуатации, так и их изменение после выполнения ремонтных и наладочных работ.

Верхняя полоса экрана – **календарь**, который позволяет выбрать период времени для анализа. Прокрутка и масштабирование календаря осуществляются колесом мыши, масштабирование – с удержанием Ctrl. Выбор диапазона дат – левой кнопкой мыши. Все данные в рабочей области отображаются за выбранные в календаре даты.

Верхняя половина рабочей области состоит из нескольких таблиц (слева направо):

- 1. Параметры** выбранного гидроагрегата – список параметров, участвующих в диагностике (таблица DiagChannels в БД конфигурации);
- 2. Нарботка** по режимам – время в часах, отработанное гидроагрегатом при данном значении напора и активной мощности (шаг сетки задаётся в настройках);
- 3. Моменты работы** – список минут, в которые гидроагрегат работал в выбранном в таблице 2 диапазоне режимов, с указанием значения активной мощности и напора;
- 4. Выбранные параметры** – параметры, выбранные в таблице 1, с фактическими значениями за выбранный момент работы в таблице 3.

В нижней половине рабочей области может быть отображён **график** изменения выбранного параметра (таблица 4) во времени за те моменты, которые гидроагрегат работал в выбранном диапазоне режимом (таблица 2). Для отображения графика нужно перетащить мышью строку из таблицы Выбранные параметры в область графика. При запросе данных за большой отрезок времени (несколько месяцев и более) может быть значительная задержка отображения данных.

График отображается кусочным и содержит только точки, соответствующие выбранному режиму работы. Устойчивая тенденция изменения значений или скачкообразное изменение во

время ремонтных работ могут говорить о существенном изменении вибрационного состояния гидроагрегата.

4.7 Экран «События»

Экран отображает события за последние сутки. События формируются контроллерами и сохраняются в базе данных сервером виброконтроля. При наличие новых событий, сервер уведомляет клиентов, и клиенты читают из базы данных все события на глубину 1 сутки от текущего времени. Чтение событий обеспечивает RawServer.

Каждое событие содержит следующую информацию:

- **Время** – дата и время события (присваивается контроллером);
- **ГА** – номер гидроагрегата;
- **Группа** – тип источника события;
- **Устройство** – определенное устройство или сигнал, послужившие источником события;
- **Событие** – описание события.

События отображаются для текущего выбранного гидроагрегата или для всех гидроагрегатов. В правой части экрана расположен фильтр событий, представляющий собой дерево с уровнями: Гидроагрегат, Группа, Устройство. В фильтре может быть включено или отключено отображение как отдельного устройства, так и любой ветви целиком.

4.8 Экран «Настройка»

На данном экране собраны вспомогательные экраны настройки и самодиагностики системы. В меню можно выбрать один из следующих экранов:

1. **Конфигурирование** – отображает структуру контроллеров в системе (планируется реализация экрана конфигурирования состава системы).
2. **Настройка** – константы для задач диагностики (таблица Consts).
3. **Самодиагностика** – системные параметры состояния каждого из контроллеров, серверных плагинов и других источников данных.

5 ЖИЗНЕННЫЙ ЦИКЛ ПО

5.1 Требования к эксплуатационному персоналу

Персонал, участвующий в эксплуатации ПО, может выступать в роли пользователя ПО или администратора ПО, и, в зависимости от роли, должен иметь соответствующую квалификацию.

Пользователь – специалист, осуществляющий анализ данных с использованием ПО. Требования к пользователю:

- достаточные знания в части объекта автоматизации;
- достаточные теоретические знания и практические навыки в части вибрационной диагностики электрических машин;
- прохождение обучения использованию программы или самообучение на основе документации.

Администратор – специалист, осуществляющий установку и настройку ПО, конфигурирование базы данных для обеспечения работоспособности программы. Требования к администратору:

- достаточные знания в части объекта автоматизации;
- знание структуры используемой базы данных и навыки её администрирования;
- знание принципов работы и настройки данного ПО;
- владение компьютерной грамотностью на уровне опытного пользователя;
- прохождение обучения настройке и использованию программы или самообучение на основе документации.

5.2 Устранение неисправностей

В случае нештатной работы ПО в ходе его эксплуатации, пользователь должен обратиться к администратору за консультацией и устранением возможных неисправностей в настройках ПО, базе данных и используемой сетевой инфраструктуре объекта автоматизации. При невозможности устранения неисправности силами администратора или в случае выявления неисправностей непосредственно в ПО, администратор должен обратиться к разработчику ПО, максимально точно и подробно описав суть неисправности (внешние проявления, условия возникновения и т.д.).

При необходимости, разработчиком может быть запрошена у администратора дополнительная информация (копии экрана, системные журналы, конфигурация оборудования, структура базы данных, фрагменты базы данных и т.д.), которая рассматривается разработчиком как конфиденциальная и не подлежит распространению, но, по согласованию с администратором, может долгосрочно сохраняться разработчиком для обеспечения дальнейшей технической поддержки.

Разработчик, на основе обращения администратора, проводит анализ возникшей неисправности и оказывает консультативную помощь или проводит работу по устранению неисправности в рамках договорных отношений заказчика и разработчика.

5.3 Техническая поддержка

Информацию о замечаниях и неисправностях ПО, выявленных в ходе его эксплуатации, следует направлять разработчику ПО по следующим контактным данным:

- сайт: **www.rakurs.com**;
- электронная почта: **info@rakurs.com**;
- телефон: **(812) 252-32-44**.

По этим же контактным данным можно направлять предложения по развитию и совершенствованию ПО, а также запросы на получение дополнительных лицензий на использование ПО и все другие вопросы, связанные с программой.

При поставке ПО в составе автоматизированной системы, на него распространяются гарантийные обязательства по самой автоматизированной системе. При поставке ПО отдельно, условия поддержки пользователей определяются договором.

В ходе работ по развитию ПО и устранению выявленных замечаний и неисправностей, выпускаются новые версии ПО. Обновление ПО выполняется по согласованию заказчика и разработчика, и может быть выполнено как специалистами заказчика по инструкциям, предоставляемым разработчиком, так и специалистами разработчика, как удалённо, так и с выездом на место. При этом вопросы совместимости версий и возможности обновления ПО необходимо оговаривать с разработчиком. Перечень версий ПО с указанием внесённых изменений ведётся разработчиком и может быть предоставлен заказчику для решения вопросов совместимости версий и целесообразности обновления ПО.

5.4 Жизненный цикл ПО

Поддержание жизненного цикла ПО, в том числе разработка, пуско-наладочные работы на объекте автоматизации, техническая поддержка заказчика, получение обратной связи от заказчика, регламентируются внутренними документами Системы менеджмента качества разработчика, сертифицированной на соответствие ISO 9001:2015.