

# VISOR

**Версия 5.3**

**Описание**

Рев. 02В-01/10

<b>ВВЕДЕНИЕ .....</b>	<b>3</b>
<b>1. ОБЗОР.....</b>	<b>4</b>
1.1. ОБОРУДОВАНИЕ .....	4
1.2. СБОР ДАННЫХ .....	5
1.3. РЕЖИМЫ СБОРА ДАННЫХ .....	7
1.3.1. Режим “TIMESLOT”.....	7
1.3.2. Режим “TIMESLOT_SIMPLE”.....	8
1.3.3. Режим “SYNCHRONOUS” .....	8
<b>2. КОММУНИКАЦИОННЫЙ СЕРВЕР VISOR_ON.....</b>	<b>10</b>
2.1. СТРУКТУРА БАЗЫ ДАННЫХ .....	10
2.2. КОНФИГУРИРОВАНИЕ .....	11
2.2.1. Каналы (Channel settings) .....	11
2.2.2. Опции (Options).....	13
2.2.3. Ведение логов (Logs) .....	13
<b>3. КЛИЕНТ-ПРОСМОТРИТЕЛЬ VISOR.....</b>	<b>14</b>
3.1. ТЕГИ .....	15
3.1.1. Структура тегов .....	15
3.1.2. Создание и редактирование тегов .....	16
3.1.3. Группы тегов .....	17
3.1.4. Управление тегами.....	17
3.2. ДИАГРАММА.....	18
3.3. ТАБЛИЦА .....	20
3.4. ПОИСК .....	21
3.5. АВТОНОМНЫЕ ФАЙЛЫ .....	22
<b>4. ПРОЦЕДУРА УСТАНОВКИ.....</b>	<b>23</b>
4.1. СОСТАВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ .....	23
4.2. ПОСЛЕДОВАТЕЛЬНОСТЬ УСТАНОВКИ .....	23
<b>5. USB КЛЮЧ АППАРАТНОЙ ЗАЩИТЫ VISOR-KEY .....</b>	<b>25</b>
<b>6. ПОДДЕРЖКА И КОНТАКТЫ .....</b>	<b>26</b>

## ВВЕДЕНИЕ

Для повышения производительности и качества продукции производственные технологии совершенствуются и усложняются. Достигается это применением всё более сложных и интеллектуальных автоматизированных систем управления. Увеличивается количество датчиков и исполнительных устройств, возрастают объём информации и скорость её обработки, усложняются алгоритмы и аппаратные средства автоматизации - всё это затрудняет наладку и поиск неисправностей. В связи с этим возникает необходимость наблюдать не только текущие данные, но и иметь возможность осуществлять поиск и изучение данных за прошедший период времени. Традиционные инструменты, такие как осциллограф или самописец, ограничены в возможностях применения, так как количество сигналов огромно и они распределены по производственному цеху или установке.

На помощь приходит сама система автоматизации, которая измеряет и использует в реализации технологии практически всю необходимую информацию об объекте производства. Чтение и сохранение этой информации позволяет решать следующие задачи:

- качественная наладка систем автоматизации;
- наблюдение за технологическими параметрами производства;
- оптимизация алгоритмов управления технологическим процессом;
- определение неправильного функционирования и неполадок в работе оборудования;
- контроль действий операторов и обслуживающего персонала.

Требования, предъявляемые к таким системам чтения и хранения данных, следующие:

- высокая скорость чтения и записи данных технологического процесса;
- многоканальность;
- хранение данных в компактном виде;
- достаточное количество переменных;
- быстрый доступ к хронологической базе данных.

Программное обеспечение "**VISOR**" позволяет реализовать вышеперечисленные задачи и обеспечивает непрерывную высокоскоростную регистрацию данных технологического процесса посредством имеющейся системы автоматизации на базе PLC и персонального компьютера. При этом не требуется никакого дополнительного оборудования.

## 1. ОБЗОР

Программное обеспечение “**VISOR**” содержит следующие приложения:

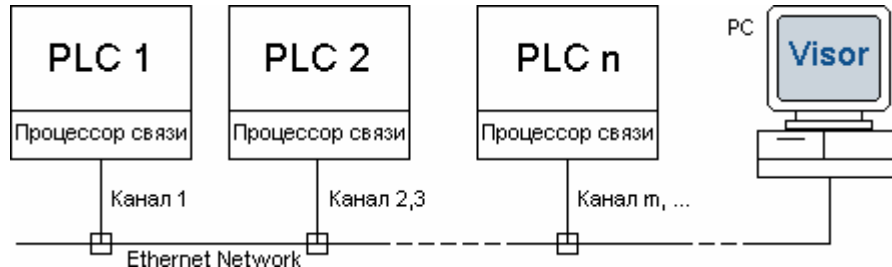
- **Visor\_On** - коммуникационный сервер, осуществляющий чтение данных из программируемых логических контроллеров (PLC), упаковку и сохранение их в базе данных на носителе;
- **Visor** - клиент-просмотрщик, который обеспечивает прямой или удаленный доступ к базе сохранённых данных, навигацию, отображение и печать их в виде диаграмм и таблиц, а также наблюдение переменных процесса в реальном времени; кроме того, он позволяет сохранять интересующий фрагмент данных в виде автономного файла, для просмотра которого достаточно наличие только просмотрщика.

Оба приложения независимы друг от друга и функционируют автономно. Поток данных представляет собой однонаправленную схему:

**Данные Процесса -> PLC -> Visor\_On -> База Данных -> Visor**

### 1.1. ОБОРУДОВАНИЕ

Типовая структура организации сбора данных с использованием “**VISOR**” показана на Рисунке 1-1.



**Рисунок 1-1.** Типовая структура организации сбора данных

Рабочая станция (PC) с предустановленным и сконфигурированным программным обеспечением “**VISOR**” подключается к системе автоматизации (PLC) посредством “*Ethernet*” сети.

Детализация хронологических данных (частота выборок) напрямую зависит от производительности сети в целом. Но для определенного количества данных, “**VISOR**” обеспечивает чтение данных синхронно с циклом исполнения программы PLC (каждый цикл), в режиме **“SYNCHRONOUS”**.

Программное обеспечение “**VISOR**” устанавливается на IBM-совместимом персональном компьютере (PC) с рекомендуемой минимальной конфигурацией:

- CPU Pentium IV 1600 MHz;
- 1024MB RAM;
- жесткий диск 120GB;
- видеосистема с разрешением 1024x768, RAM video 16MB;
- Ethernet сетевая карта.

Операционные системы, поддерживаемые “**VISOR**”:

- Windows 2000 Pro/Server;
- Windows XP;
- Windows 2003 Server;
- Windows Vista.

Поддерживаемые типы PLC:

- Siemens SIMATIC S7/C7.

## 1.2. СБОР ДАННЫХ

Чтобы коммуникационный сервер **Visor\_On** мог читать данные, их нужно предварительно подготовить в PLC. Для этого необходимо создать блок данных и обеспечить запись в него требуемых переменных.

Данные читаются частями, максимальный размер которых - 2048 байт. Для чтения части данных из PLC коммуникационный сервер **Visor\_On** использует канал. Каналы независимы, каждый канал конфигурируется индивидуально. Для каждого PLC можно иметь один и более каналов, чем достигается пропорциональное увеличение размера читаемых данных. Максимальное количество каналов - 30 (зависит от лицензии). Таким образом, коммуникационный сервер может читать до 61440 байт данных, при этом число PLC может достигать 30.

Основные варианты организации сбора и анализа данных представлены на блок-схеме изображенной на Рисунке 1-2.

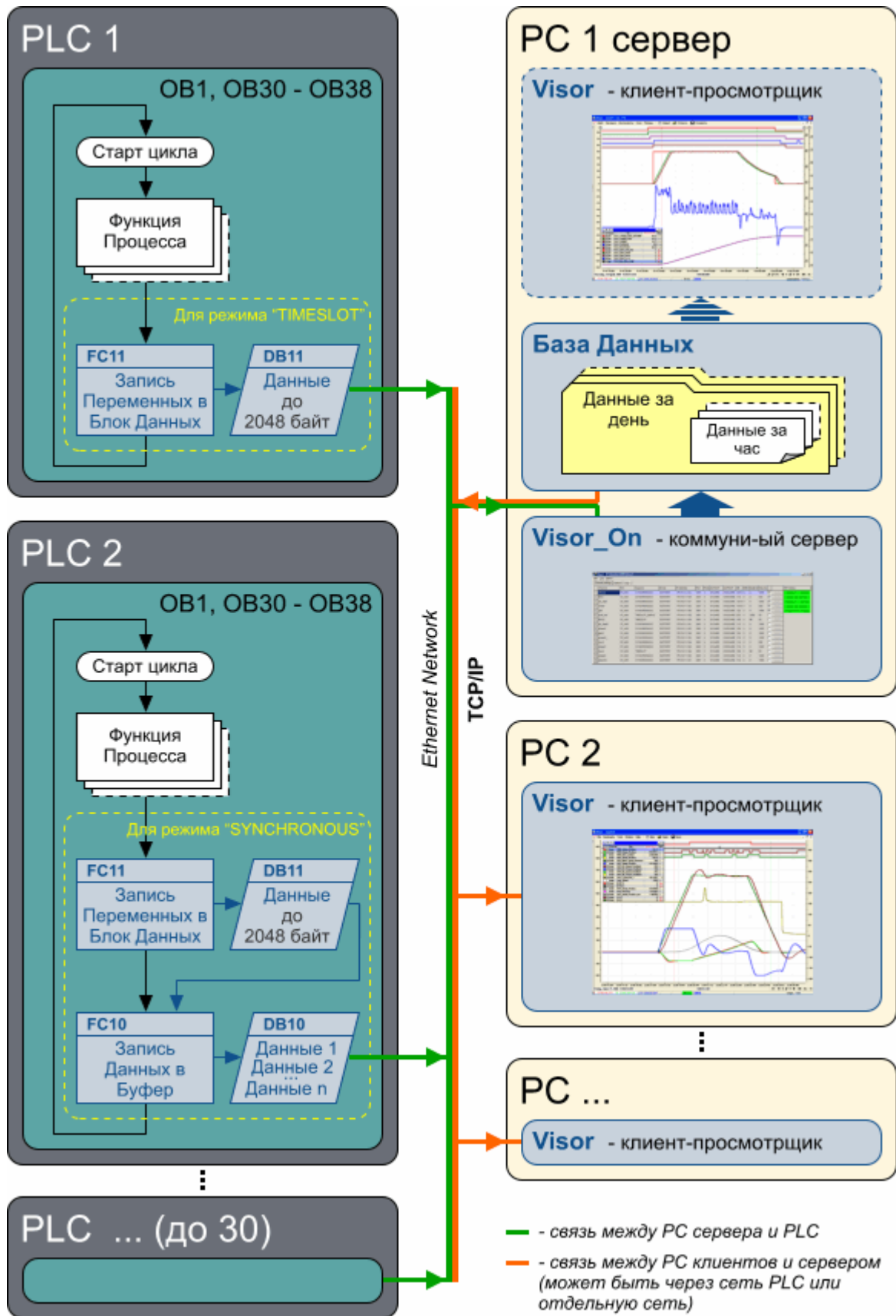


Рисунок 1-2. Варианты организации сбора и анализа данных

### 1.3. РЕЖИМЫ СБОРА ДАННЫХ

“VISOR” поддерживает 3 режима чтения данных из PLC:

- TIMESLOT - с фиксированным периодом выборок;
- TIMESLOT\_SIMPLE - с фиксированным периодом выборок, упрощенный;
- SYNCHRONOUS - синхронный с циклом PLC.

#### 1.3.1. РЕЖИМ “TIMESLOT”

Это режим прямого чтения данных. Для режима “TIMESLOT” блок данных, помимо переменных, должен иметь заголовок, содержащий время PLC. Время PLC можно получить с помощью стандартных функций процессора (CPU). Формат представления времени в заголовке блока совпадает с форматом “DATE\_AND\_TIME” PLC SIMATIC S7 и имеет размер 8 байт:

- |                   |                     |                               |
|-------------------|---------------------|-------------------------------|
| • Байт 0:         | год                 | 90H ... 89H (от 1990 до 2089) |
| • Байт 1:         | месяц               | от 01H до 12H                 |
| • Байт 2:         | день                | от 01H до 31H                 |
| • Байт 3:         | час                 | от 00H до 23H                 |
| • Байт 4:         | минута              | от 00H до 59H                 |
| • Байт 5:         | секунда             | от 00H до 59H                 |
| • Байт 6:         | десятки миллисекунд | от 00H до 99H                 |
| • Ст. полубайт 7: | миллисекунда        | от 0H до 9H                   |
| • Мл. полубайт 7: | день недели         | от 1H до 7H (не используется) |

Все байты должны быть в формате BCD согласно спецификации Siemens.

Коммуникационный сервер **Visor\_On** периодически, через равные параметрируемые промежутки времени, производит чтение данных из PLC и записывает их в базу хронологических данных на рабочей станции (PC). Период чтения данных выбирается исходя из желаемой детализации данных. Если PLC не способен предоставить данные в течение установленного интервала времени, то период чтения увеличивается автоматически. Обычно в этом режиме не удаётся достичь наивысшей детализации (читать данные каждый цикл исполнения программы), так как это требует высокой пропускной способности канала передачи данных, что иногда физически не возможно. Этот режим следует использовать, если не требуется вести запись быстроизменяющихся переменных.

Поток данных в этом режиме:

**Данные Процесса -> Блок Данных PLC -> Visor\_On -> База Данных**

Преимущества режима “TIMESLOT”:

- простота использования и конфигурирования;
- не требуется никакого дополнительного специального программного обеспечения в PLC.

Недостатки режима “TIMESLOT”:

- низкая детализация данных для быстрых программных циклов, когда время цикла значительно меньше периода чтения данных;
- уменьшение периода чтения данных из PLC влечёт за собой увеличение нагрузки на сетевое оборудование.

### 1.3.2. РЕЖИМ “TIMESLOT\_SIMPLE”

Это самый простой режим сбора данных. Режим “TIMESLOT\_SIMPLE” отличается от “TIMESLOT” тем, что время каждого чтения блока данных берётся с внутренних часов рабочей станции (PC). В этом режиме заголовок в блоке данных, содержащий время PLC, не требуется.

Преимущества режима “TIMESLOT\_SIMPLE”:

- те же, что в режиме “TIMESLOT”.

Недостатки режима “TIMESLOT\_SIMPLE”:

- те же, что в режиме “TIMESLOT”;
- так как используется время с внутренних часов рабочей станции, то происходит некоторая рассинхронизация реального времени изменения данных и времени их получения, это смещение происходит из-за задержки на транспортировку данных по сети, которая переменна.

### 1.3.3. РЕЖИМ “SYNCHRONOUS”

В этом режиме происходит чтение данных не непосредственно из блока данных переменных процесса, а из блока данных циклического буфера. Заполнение буфера происходит синхронно с циклом PLC посредством специального программного обеспечения, написанного на языке используемого PLC. Поток данных в этом случае выглядит следующим образом:

**Данные Процесса -> Блок Данных PLC -> Блок Данных Буфера PLC -> Visor\_On -> База Данных**

Благодаря использованию буфера, нет необходимости часто читать данные из PLC, нагружая сетевое оборудование. Кроме того, за счет буферизации достигается возможность чтения данных синхронно с циклом исполнения программы PLC (каждый цикл). Производительность данного режима зависит от:

- производительности и загруженности коммуникационного процессора PLC;
- количества переменных (объём данных);
- скорости изменения переменных.

В Таблице 1-1 приводятся примеры производительности режима для PLC S7-400 при различных временах цикла исполнения программы, значения переменных читаются в каждом цикле.

**Таблица 1-1. Примеры производительности режима “SYNCHRONOUS”**

Цикл PLC, мс	Частота изменения данных	Размер буфера в PLC, КБайт	Периодичность чтения буфера из PLC, мс	Объём читаемых данных, Байт	Поток данных, КБайт/с
2	каждый цикл	16	500	40	~ 18 - 20
2	каждый 5-й цикл	16	500	200	~ 16 - 18
10	каждый цикл	16	500	200	~ 16 - 18
10	каждый 10-й цикл	16	1000	1000	~ 8 - 10
20	каждый цикл	16	500	400	~ 8 - 9
20	каждый 10-й цикл	16	1000	2000	~ 4 - 5



Преимущества режима очевидны:

- синхронность чтения данных с их изменением (чтение синхронно с циклом PLC);
- гарантирована максимальная детализация данных;
- малая нагрузка на сетевое оборудование.

К недостаткам можно отнести:

- необходим дополнительный объём свободной памяти для циклического буфера в PLC.

**Внимание!** Режим “SYNCHRONOUS” является более совершенным, и рекомендуется к приоритетному использованию.

## 2. КОММУНИКАЦИОННЫЙ СЕРВЕР VISOR\_ON

Назначение коммуникационного сервера - это чтение данных из программируемых логических контроллеров (PLC), упаковка и сохранение их в базе данных на носителе.

### 2.1. СТРУКТУРА БАЗЫ ДАННЫХ

Физически база данных представляет собой папку с именем *"DATA"*, располагающуюся в рабочей директории приложения. Её содержимое - это вложенные папки с именами, состоящими из даты:

*"YY\_MM\_DD"* - год\_месяц\_день.

Таким образом, каждая папка содержит в себе данные за один день.

Внутри каждой такой папки хранятся файлы, организованные в почасовом порядке отдельно для каждого канала набором файлов с расширениями *".dat"*, *".pnt"*, *".min"*. Имя файла состоит из часа и имени канала:

*"hh\_xxxxxx"* - час\_имя канала.

Пример организации базы данных представлен на Рисунке 2-1.

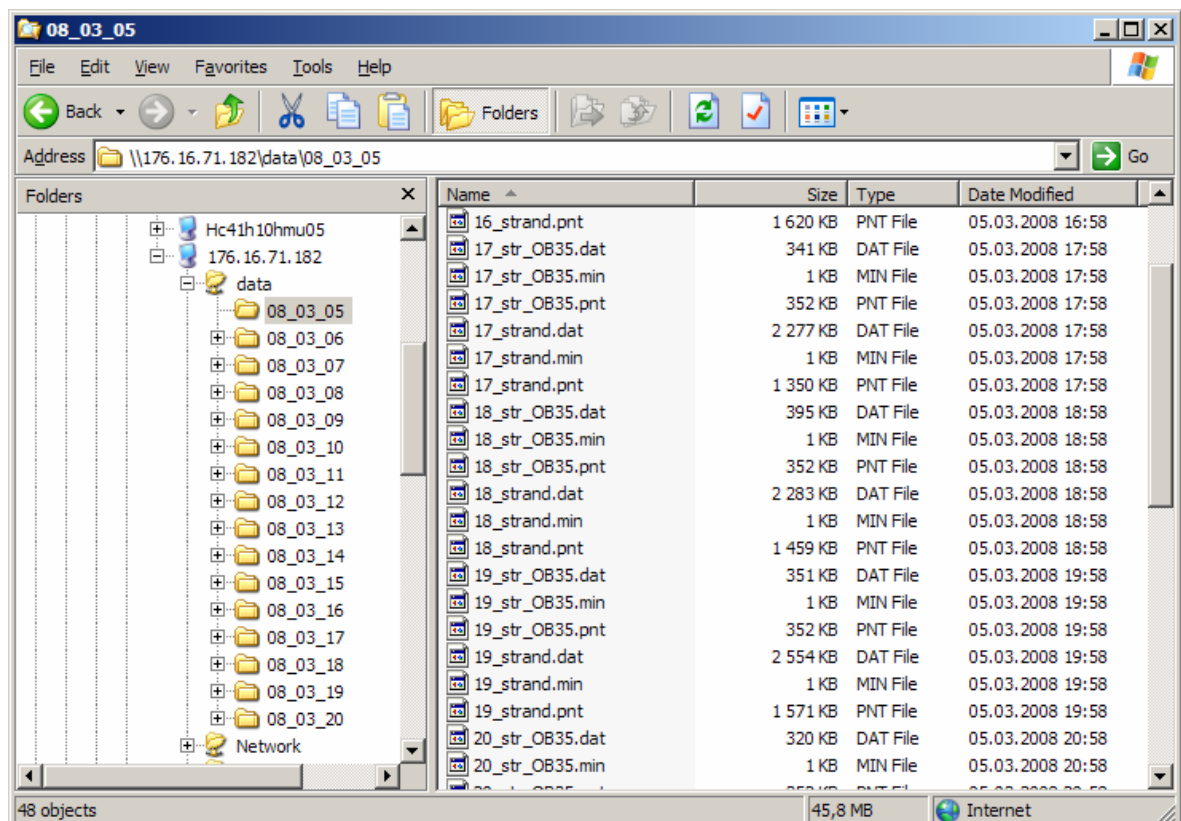


Рисунок 2-1. Пример структуры базы данных

## 2.2. КОНФИГУРИРОВАНИЕ

Окно приложения **Visor\_On** содержит страницы, выбираемые закладками:

- “*Channel settings*” - таблица конфигурации каналов;
- “*Options*” - опции;
- “*Logs*” - протокол сообщений и ошибок.

### 2.2.1. КАНАЛЫ (CHANNEL SETTINGS)

Для запуска канала необходимо его описать в строке таблицы каналов и активировать. Пример конфигурации канала представлен на Рисунке 2-2.

Channel	PLC	Capture	Driver	IP address	Port	Ports	locTSAP	remTSAP	DB	DBB	length, b	time, ms		NET status	
1 strand	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.181	3001	3	01CARD	0103CARD	1018	0	0	1000	<input checked="" type="checkbox"/>	Apply	1000ms 0,1 1369b/s
2 BOC	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.184	3001	3	01CARD	0103CARD	116	0	0	500	<input checked="" type="checkbox"/>	Apply	500ms 0,2 5371b/s
3 str_fast	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.181	3001	3	01CARD	0103CARD	1019	0	0	1000	<input checked="" type="checkbox"/>	Apply	1000ms 0,1 1487b/s
4 shear	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.183	3001	3	01CARD	0103CARD	1018	0	0	500	<input checked="" type="checkbox"/>	Apply	500ms 0,2 5280b/s
5 gen	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.180	3001	3	01CARD	0103CARD	120	0	0	1000	<input checked="" type="checkbox"/>	Apply	1000ms 0,0 1410b/s
6 soft_red	S7_400	TIMESLOT_SIMPLE	ISOTCPS7	176.16.11.182	3001	3	01CARD	0103CARD	220	0	1200	50	<input type="checkbox"/>	Apply	
7 BOC2	S7_400	TIMESLOT	ISOTCPS7	176.16.11.184	3001	3	01CARD	0103CARD	108	0	96	30	<input type="checkbox"/>	Apply	
8 str_fast2	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.181	3001	3	01CARD	0103CARD	110	0	0	1000	<input type="checkbox"/>	Apply	
9 shear2	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.183	3001	3	01CARD	0103CARD	109	0	0	1000	<input type="checkbox"/>	Apply	
10 gen2	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.180	3001	3	01CARD	0103CARD	120	0	0	1000	<input type="checkbox"/>	Apply	
11 shear2_	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.183	3001	3	01CARD	0103CARD	119	0	0	1000	<input type="checkbox"/>	Apply	
12 boc3	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.184	3001	3	01CARD	0103CARD	119	0	0	500	<input type="checkbox"/>	Apply	
13 shear3	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.183	3001	3	01CARD	0103CARD	119	0	0	1000	<input type="checkbox"/>	Apply	
14 boc4	S7_400	TIMESLOT	ISOTCPS7	176.16.11.184	3001	3	01CARD	0103CARD	108	0	96	50	<input type="checkbox"/>	Apply	
15 shear4	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.183	3001	3	01CARD	0103CARD	119	0	0	1000	<input type="checkbox"/>	Apply	
16 strand3	S7_400	SYNCHRONOUS	ISOTCPS7	176.16.11.181	3001	3	01CARD	0103CARD	119	0	0	1000	<input type="checkbox"/>	Apply	

**Рисунок 2-2.** Окно приложения **Visor\_On** на странице “*Channel settings*”

Описание полей конфигурации канала смотрите в Таблице 2-1.

**Таблица 2-1.** Описание полей конфигурации канала

Наименование	Описание
Channel	Имя канала. Длина имени канала допускается не более 20 символов, без пробелов и вспомогательных знаков, запрещенных в использовании в именах файлов операционной системы.
PLC	Тип PLC. В настоящее время поддерживаются "Siemens": S7-200; S7-300; S7-400.
Capture	TIMESLOT / TIMESLOT_SIMPLE / SYNCHRONOUS, режимы сбора данных.
Driver	ISOTCPS7 / TCPS7, протокол обмена с PLC "Simatic S7". ISOTCPS7 - родной "Siemens" протокол непосредственного чтения данных из процессора (CPU); TCPS7 - это т.н. TCP протокол FETCH/WRITE, при этом в "Hardware configuration" PLC создается TCP соединение FETCH-passive на определённый порт.
IP address	IP адрес PLC.
Port	Стартовый номер порта связи с PLC (только для протокола TCPS7).
Ports	Количество доступных портов в PLC. Параметр показывает, сколько всего имеется доступных портов, начиная со стартового порта (только для протокола TCPS7).
locTSAP	Локальный TSAP. В настоящий момент не используется и зарезервирован для совместимости с PLC "Siemens S5".
remTSAP	Удаленный TSAP. Здесь кодируется позиция CPU Siemens S7, XXYYCARD: <ul style="list-style-type: none"> <li>• XX - номер стойки;</li> <li>• YY - позиция CPU в стойке.</li> </ul> Номера назначаются согласно "Hardware configuration" PLC.
DB	Номер блока данных для чтения. Для режимов TIMESLOT и TIMESLOT_SIMPLE указывается номер блока данных переменных. Для режима SYNCHRONOUS указывается номер блока данных буфера.
DBB	Адрес первого байта читаемых данных блока данных. Рекомендуется, во избежание путаницы с адресами тегов, назначать этот параметр всегда равным нулю. Для режима SYNCHRONOUS этот параметр не используется.
Length	Длина читаемых данных в байтах. Для режима SYNCHRONOUS этот параметр не используется.
Time	Период чтения данных из PLC, мсек.
NET status	Отображает текущее состояние канала.

**Внимание!** Не следует назначать параметр "Time" слишком коротким, это нагружает процессоры PLC и сетевое оборудование.

### 2.2.2. Опции (OPTIONS)

Здесь можно установить длительность хранения данных в базе, с целью автоматической очистки жесткого диска от устаревшей информации. Длительность хранения указывается в сутках. Если указать “0”, то устаревшие данные удаляться не будут.

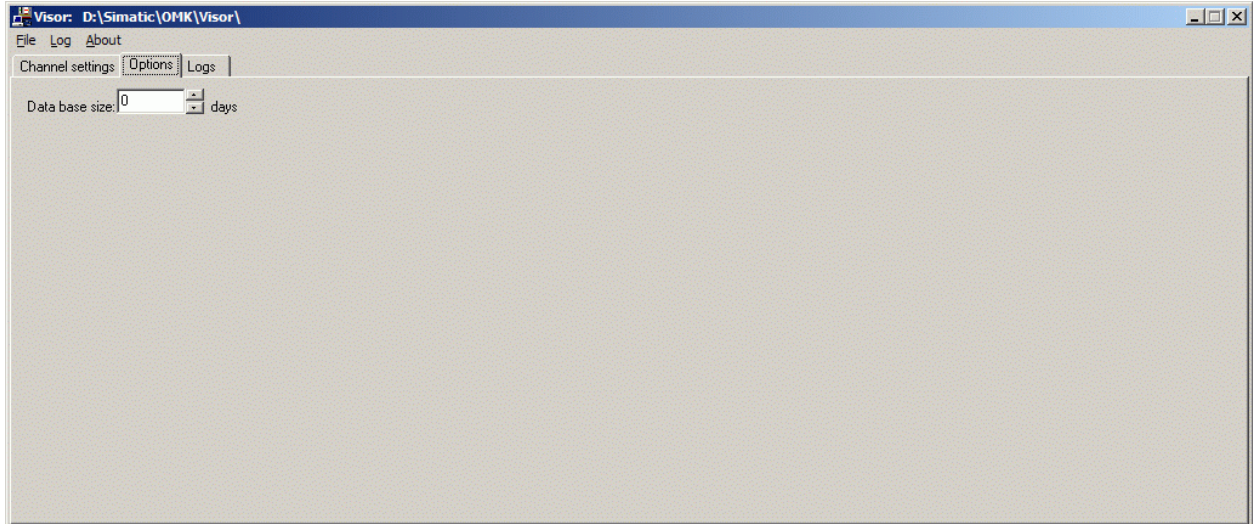


Рисунок 2-3. Окно приложения Visor\_On на странице “Options”

### 2.2.3. ВЕДЕНИЕ ЛОГОВ (LOGS)

В приложении предусмотрено ведение протоколов сообщений и ошибок. Сообщения, содержащие в своем теле метку из трех символов “#”, являются ошибками и означают потерю данных.

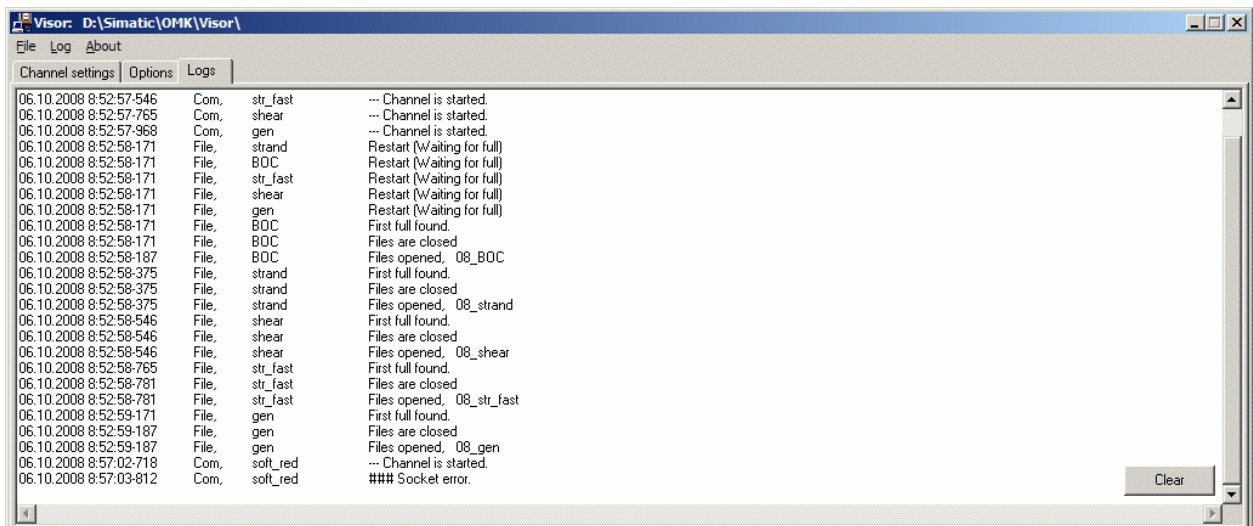


Рисунок 2-4. Окно приложения Visor\_On на странице “Logs”

Все сообщения и ошибки разбиваются по суткам и записываются в файлы с расширением “.log” в папке “DATA”.

### 3. КЛИЕНТ-ПРОСМОТРИК VISOR

Клиент-просмотрщик обеспечивает доступ к базе сохранённых данных, навигацию, отображение их в виде диаграмм и таблиц, вывод на печать, а также наблюдение переменных процесса в реальном времени.

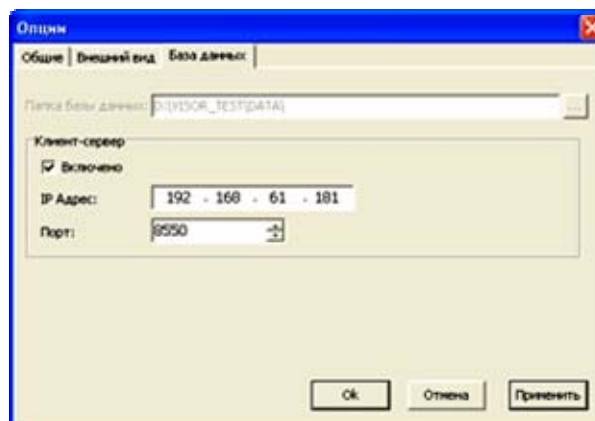
Существуют три режима работы приложения:

1. с базой хронологических данных расположенной в рабочей директории;
2. с базой хронологических данных расположенной на удалённом компьютере;
3. с автономными файлами (только диаграммы).

В первом случае клиент-просмотрщик **Visor** располагается в одной директории с коммуникационным сервером **Visor\_On**. **Visor** использует базу хронологических данных, расположенную в папке *"DATA"*, которую непосредственно формирует коммуникационный сервер.

Во втором случае клиент-просмотрщик **Visor** устанавливается отдельно, и пользователь получает доступ к базе хронологических данных, формируемой коммуникационным сервером **Visor\_On** на удалённом компьютере, через сеть. В этом случае получается многопользовательская структура, с одним сервером и множеством клиентов.

Режим работы можно выбрать в меню *«Инструменты» - «Опции» - «База данных»*.



**Рисунок 3-1.** Окно *«Опции»* приложения **Visor** на странице *«База данных»*

Для работы с базой хронологических данных необходимы списки тегов и группы тегов, располагающиеся в папках *"DBV"* и *"GRV"* в рабочей директории в виде файлов с расширением *“.dbv”* (списки тегов) и *“.grv”* (группы тегов).

Для работы в автономном режиме достаточно только автономного файла с расширением *“.visor”*, содержащего вырезку данных для выбранных тегов за определённый промежуток времени.

Переход из режима просмотра автономного файла в режим работы с базой данных происходит в момент выбора интервала времени, выходящего за пределы данных, записанных в автономном файле. В окне появившегося предупреждения можно нажатием соответствующих кнопок либо остаться в диапазоне данных, сохранённых в автономном файле, либо покинуть его.

Текущий режим индицируется на статусной панели внизу окна.

### 3.1. ТЕГИ

Перед использованием переменных их необходимо описать. Тег представляет собой описание отдельно взятой переменной в блоке данных.

#### 3.1.1. СТРУКТУРА ТЕГОВ

Тег состоит из строки содержащей несколько параметров разделяемых символом «;». Выглядит это следующим образом:

**Формат\_и\_Адрес; Имя; Тип; [Множитель]; [Комментарий]**

Параметры, заключенные в квадратные скобки, являются необязательными. В Таблице 3-1 перечислены **«Форматы»** и **«Типы»** представления переменных на экране.

**Таблица 3-1.** Форматы и типы представления переменных на экране

Размер	Формат	Тип	Представление	Описание
Бит	DBX	BOOL	Бит	0 ÷ 1
Байт	DBB	INT	Целое	-128 ÷ +127
		UINT	Беззнаковое целое	0 ÷ 255
		HEX	Шестнадцатеричное	00 ÷ FF
		BIN	Двоичное	00000000 ÷ 11111111
		CHAR	Символьное	1 ASCII символ
Слово	DBW	INT	Целое	-32768 ÷ +32767
		UINT	Беззнаковое целое	0 ÷ 65535
		HEX	Шестнадцатеричное	0000 ÷ FFFF
		BIN	Двоичное	00000000 00000000 ÷ 11111111 11111111
		CHAR	Символьное	2 ASCII символа
Двойное слово	DBD	INT	Целое	- 2147483648 ÷ + 2147483647
		UINT	Беззнаковое целое	0 ÷ 4294967295
		HEX	Шестнадцатеричное	0000 0000 ÷ FFFF FFFF
		BIN	Двоичное	00000000 00000000 ÷ 11111111 11111111 11111111 11111111
		CHAR	Символьное	4 ASCII символа
		REAL	Вещественное	+1.175495e-38 ÷ +3.402823e+38, -1.175495e-38 ÷ -3.402823e+38

**«Адрес»** - адрес переменной в блоке данных.

**«Имя»** - символическое имя переменной, длиной до 24 символов.

**«Множитель»** - это коэффициент, на который будет умножено значение переменной. По умолчанию он равен 1, и может иметь значения: 0.001, 0.01, 0.1, 1, 10, 100, 1000. Множитель имеет смысл только для типов "INT", "UINT" и "REAL". Использование этого коэффициента удобно для обратного преобразования порядка переменной, если оно было сделано в PLC.

**«Комментарий»** - описание переменной, длиной до 79 символов.

Примеры тегов:

```

DBX10.0 ;Q10.0 ;BOOL ; ;Клапан
DBD250 ;СКОРОСТЬ ;REAL ;1 ;Скорость, мм/мин
DBD344 ;ЭНКОДЕР ;UINT ;1 ;Счетчик энкодера, имп.
DBW845 ;ВЕРТ_ПОЗИЦИЯ ;INT ;0.1 ;Вертикальная позиция, мм
    
```

### 3.1.2. СОЗДАНИЕ И РЕДАКТИРОВАНИЕ ТЕГОВ

Каждому каналу соответствует свой список тегов. Теги можно создавать в приложении **Visor**. Для этого имеется встроенный редактор тегов, который можно вызвать из контекстного меню, щелкнув правой кнопкой мыши по списку тегов. Теги представляются в виде таблицы и имеют параметры описанные в [п.3.1.1](#).

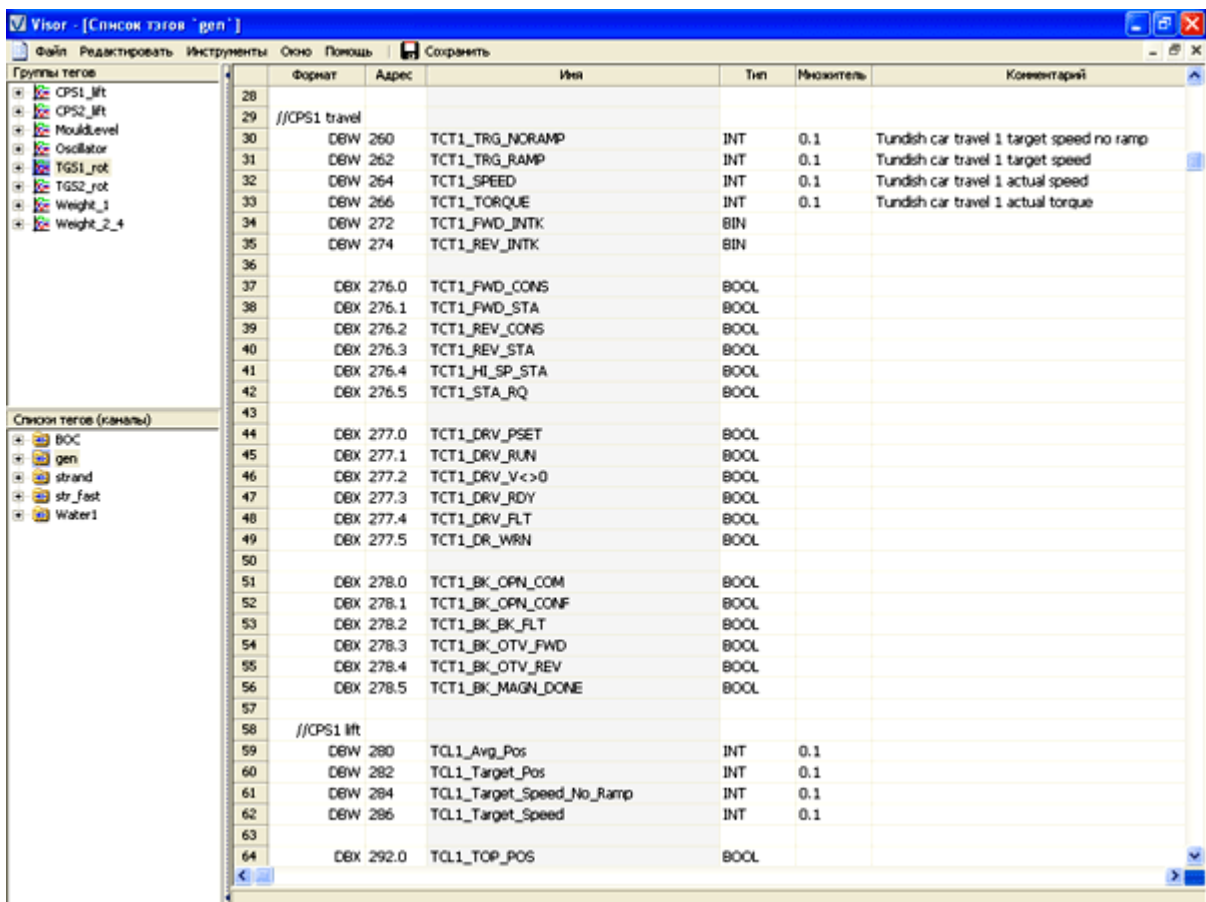


Рисунок 3-2. Окно приложения **Visor** с открытым редактором тегов

Списки тегов хранятся в файлах с расширением “*dbv*” в текстовом формате и размещаются в папке “*DBV*” в рабочей директории “*VISOR*”. Имя файла – это “*имя\_канала*”.



### 3.1.3. Группы тегов

Для удобства работы теги могут быть объединены в группы. Группирование тегов позволяет создавать наборы часто используемых переменных и оперативно открывать их в окне просмотра. Группы тегов создаются в приложении **Visor** путём набора интересующих тегов из списков и сохраняются в файлы с расширением “.grv” в папке “GRV” внутри рабочей директории “VISOR”. Группа может содержать теги из различных списков. Максимальное количество тегов в группе - 50.

### 3.1.4. УПРАВЛЕНИЕ ТЕГАМИ

В левой части главного окна приложения расположена выдвижная панель управления тегами. Панель разделена на две части: в верхней содержатся группы тегов, в нижней - каналы с соответствующими им списками тегов. Теги и их группы выбираются простым перетаскиванием мышью в поле диаграммы (“drag-and-drop”).

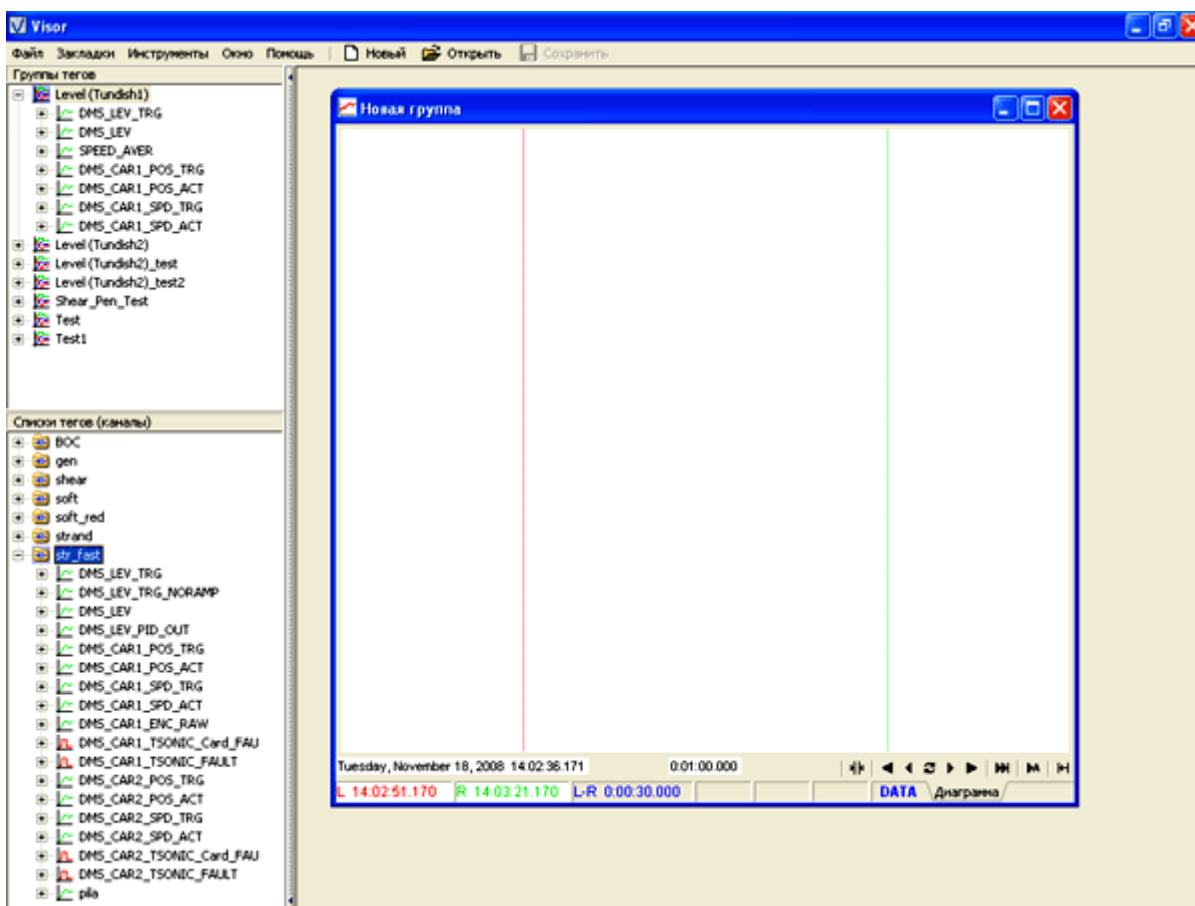


Рисунок 3-3. Окно приложения **Visor** с раскрытой панелью управления тегами

## 3.2. ДИАГРАММА

Закладка «*Диаграмма*» содержит страницу, где можно создавать окна, в которых переменные отображаются в виде графиков во времени.

Исходными параметрами времени являются стартовое время и интервал времени в окне. В нижней части окна диаграммы имеются все необходимые элементы для навигации во времени.

**Внимание!** Не следует назначать слишком длинный интервал времени, так как количество информации может оказаться очень большим, вследствие чего извлечение данных из базы и прорисовка графиков может занять много времени, вызывая у пользователя ложное мнение о «зависании» программы.

Список тегов, отображаемых на графиках, расположен в окне «*Легенда*», в котором можно настраивать свойства отображения переменных на экране. Максимальное количество тегов в окне - 50.

В распоряжении имеются десять вертикальных осей (пять слева, пять справа) и одна ось времени. Оси назначаются переменным в окне «*Легенда*». Битовые переменные отображаются вне вертикальных осей, размещаются на диаграмме сверху вниз и подсвечиваются фоном с цветом сетки.

Имеются два курсора: левый (основной) и правый (вспомогательный). Значение переменных в позиции курсоров отображается в окне «*Легенда*» в соответствующих колонках. Позиция курсоров во времени отображается в окне «*Легенда*» и статусной панели. Есть возможность наблюдать в окне «*Легенда*» разницу значений переменных и времени в различных позициях левого и правого курсоров. Левый и правый курсоры не могут меняться местами.

Для управления курсорами, настройки осей, масштабирования и смещения изображения используется мышь. Возможны следующие манипуляции:

- двойной клик левой клавишей в окне устанавливает левый курсор в позицию указателя мыши;
- двойной клик правой клавишей в окне устанавливает правый курсор в позицию указателя мыши;
- все оси могут передвигаться мышью. В комбинации с кнопками клавиатуры “*Alt*”, “*Ctrl*” и “*Shift*” возможно различное поведение осей: “*Alt*” - ускоряет перемещение оси; “*Ctrl*” - изменяет нижний предел; “*Shift*” - изменяет верхний предел;
- удерживая правую кнопку мыши в окне, можно перемещать всю диаграмму (за исключением битовых кривых по вертикали). Состояние смещения индицируется на статусной панели внизу окна;
- выделение области в окне левой кнопкой мыши сверху вниз слева направо производит масштабирование всей диаграммы (за исключением битовых кривых по вертикальной оси). Выделение в обратном направлении отменяет масштабирование и смещение (откат). Состояние масштабирования индицируется на статусной панели внизу окна.

**Пояснение!** Масштабирование и смещение в окне - это возможность приближения/удаления и перемещения диаграммы указателем мыши без навигации по времени и изменения каких-либо настроек осей. Это своего рода фокусирование на интересующей части диаграммы с возможностью последующего отката на исходную

позицию. Команды навигации во времени отменяют масштабирование и смещение.

Есть возможность наблюдать значения переменных в режиме реального времени, нажав кнопку автообновления. Периодичность обновления экрана зависит от выбранного интервала просмотра и скорости изменения переменных.

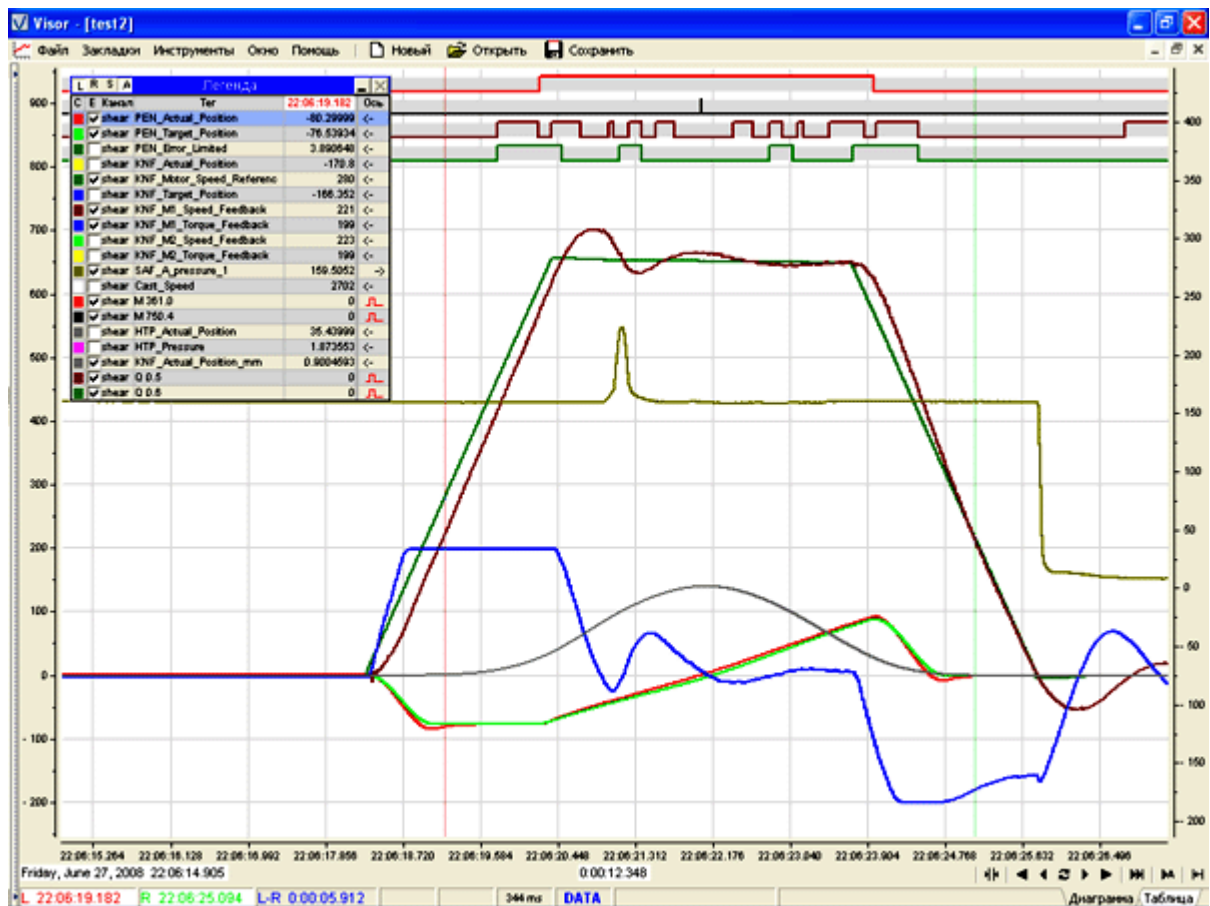


Рисунок 3-4. Окно приложения Visor на странице «Диаграмма»

### 3.3. ТАБЛИЦА

Значения переменных можно отображать в виде таблицы. Табличное представление выбирается закладкой “Таблица”, при этом вы попадаете в момент времени позиции левого курсора.

Табличное представление имеет свои особенности по сравнению с диаграммой. Таблица всегда заполняется на минуту времени. Для переключения на следующую или предыдущую минуту используется навигационная панель или быстрые клавиши.

Заполнение таблицы, в отличие от диаграммы, происходит не равномерно по времени, а по мере изменения переменных, то есть дискретизация времени между соседними рядами не постоянна и зависит от того, как часто меняются переменные.

Time	shear	PEN_Actual	PEN_Target	PEN_Error_LI	KNF_Actual	KNF_Motor	KNF_Target	KNF_M1_Spe	KNF_M1_Tor	KNF_M2_Spe	KNF_M2_Tor	SAF_A_press	Cast_Speed	M 361.0	M 750.4
22:06:19.171	-80.60999	-76.57066	4.209305	-171.1	277	-166.6057	218	199	221	199	159.2014	2703	0	0	
22:06:19.176	-80.42999	-76.57066	4.039322	-170.9	278	-166.4933	219	199	221	199	159.3027	2703	0	0	
22:06:19.181	-80.29999	-76.53934	3.890648	-170.8	280	-166.352	221	199	223	199	159.5052	2702	0	0	
22:06:19.186	-80.17999	-76.53934	3.760643	-170.7	282	-166.21	222	199	225	199	159.4039	2702	0	0	
22:06:19.190	-80.07001	-76.53934	3.640648	-170.6	283	-166.0671	223	199	226	199	159.6065	2702	0	0	
22:06:19.195	-79.98999	-76.50803	3.561981	-170.5	284	-165.9523	225	199	228	199	159.7077	2701	0	0	
22:06:19.200	-79.91998	-76.50803	3.481964	-170.4	286	-165.8081	227	199	230	199	159.4039	2701	0	0	
22:06:19.205	-79.85999	-76.47671	3.443275	-170.3	288	-165.663	229	199	231	199	159.3027	2700	0	0	
22:06:19.210	-79.81	-76.47671	3.383278	-170.1	289	-165.5171	230	199	233	199	159.3027	2700	0	0	
22:06:19.214	-79.77997	-76.47671	3.33329	-170	291	-165.3705	232	199	235	199	159.3027	2700	0	0	
22:06:19.219	-79.76996	-76.47671	3.303261	-169.9	292	-165.2527	233	199	236	199	159.2014	2700	0	0	
22:06:19.224	-79.76001	-76.44539	3.32457	-169.8	294	-165.1046	234	199	238	199	159.4039	2699	0	0	
22:06:19.229	-79.73999	-76.44539	3.314621	-169.7	295	-164.9557	237	199	238	199	159.5052	2699	0	0	
22:06:19.234	-79.70996	-76.44539	3.294601	-169.6	297	-164.806	238	199	241	199	159.2014	2699	0	0	
22:06:19.239	-79.64001	-76.44539	3.264572	-169.4	298	-164.6556	240	199	242	199	159.5052	2699	0	0	
22:06:19.243	-79.52997	-76.41408	3.225937	-169.3	300	-164.5043	240	199	244	199	159.4039	2698	0	0	
22:06:19.248	-79.44	-76.41408	3.115891	-169.2	301	-164.3828	243	199	245	199	159.5052	2698	0	0	
22:06:19.253	-79.34998	-76.41408	3.025925	-169.1	303	-164.2301	244	199	247	199	159.7077	2698	0	0	
22:06:19.257	-79.26001	-76.41408	2.935898	-169	304	-164.0767	245	199	248	199	159.7077	2698	0	0	
22:06:19.262	-79.20001	-76.41408	2.845932	-168.8	306	-163.9224	247	199	250	199	159.5052	2698	0	0	
22:06:19.267	-79.15997	-76.44539	2.754623	-168.7	307	-163.7985	249	199	252	199	159.2014	2699	0	0	
22:06:19.272	-79.12	-76.47671	2.683266	-168.5	309	-163.6428	250	199	253	199	159.1001	2700	0	0	
22:06:19.276	-79.07001	-76.50803	2.611969	-168.4	310	-163.4863	252	199	255	199	158.9988	2701	0	0	
22:06:19.281	-79.01001	-76.53934	2.530663	-168.3	312	-163.3606	254	199	256	199	159.2014	2702	0	0	
22:06:19.286	-78.94	-76.60197	2.408035	-168.2	313	-163.2027	255	199	258	199	159.5052	2704	0	0	
22:06:19.290	-78.85999	-76.63329	2.306709	-168.1	315	-163.044	257	199	259	199	159.5052	2705	0	0	
22:06:19.295	-78.76996	-76.66461	2.195374	-167.9	316	-162.8845	259	199	260	199	159.4039	2706	0	0	
22:06:19.300	-78.67999	-76.66461	2.105347	-167.8	318	-162.7564	260	199	262	199	159.4039	2706	0	0	
22:06:19.305	-78.58997	-76.69593	1.984062	-167.7	319	-162.5955	262	199	263	199	159.1001	2707	0	0	
22:06:19.309	-78.51001	-76.72726	1.862709	-167.6	321	-162.4338	264	199	265	199	159.4039	2708	0	0	
22:06:19.314	-78.44	-76.72726	1.782753	-167.4	322	-162.3039	265	199	267	199	159.3027	2708	0	0	
22:06:19.319	-78.38	-76.75857	1.681435	-167.3	324	-162.1407	267	199	268	199	159.4039	2709	0	0	
22:06:19.323	-78.32001	-76.75857	1.621437	-167.1	325	-161.9768	268	199	268	199	159.4039	2709	0	0	

Рисунок 3-5. Окно приложения Visor на странице «Таблица»

### 3.4. Поиск

Поиск используется для определения того или иного состояния переменной. Поиск можно осуществлять на диаграмме и в таблице. В режиме диаграмма окно поиска вызывается из окна «*Легенда*» при наведении указателя мыши на тег и нажатии правой кнопки. Направление поиска во времени всегда вперед.

Для битовых значений критериями поиска могут быть:

- «0»;
- «1»;
- «Передний фронт»;
- «Задний фронт».

Для вещественных значений критериями поиска могут быть:

- «Равно»;
- «Не равно»;
- «Больше»;
- «Меньше»;
- «Пределы» - принадлежность пределам.

Поиск производится для одной переменной.

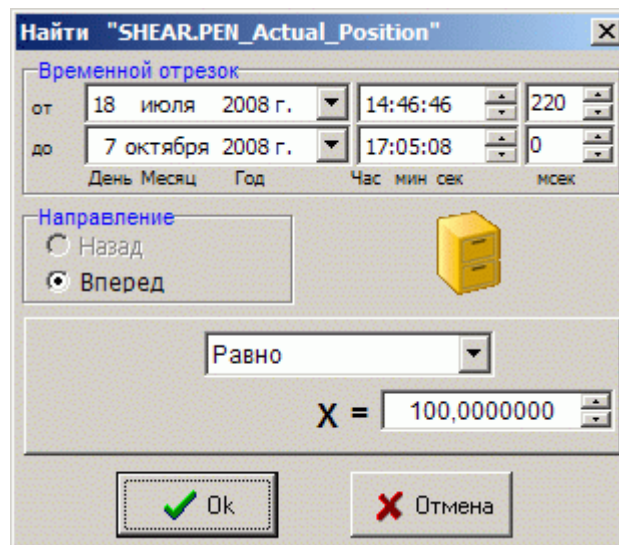


Рисунок 3-6. Окно поиска значения переменной

### 3.5. АВТОНОМНЫЕ ФАЙЛЫ

**Visor** позволяет сохранять интересуемый фрагмент данных в виде автономного файла. Файл имеет расширение *“.visor”* и содержит вырезку данных для выбранных тегов за определённый промежуток времени. Для просмотра автономного файла достаточно наличие только клиента-просмотрщика. Использование автономных файлов удобно для сохранения заинтересовавших фрагментов, а также для передачи их другим лицам для изучения. Другое полезное предназначение автономных файлов - это создание закладок для последующего быстрого доступа в нужное место в базе данных. В меню диаграммы имеются все необходимые команды для создания и открытия автономных файлов.

**Внимание!** Автономные файлы используются только для диаграммы.

## 4. ПРОЦЕДУРА УСТАНОВКИ

### 4.1. СОСТАВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В комплект программного обеспечения “VISOR” входят:

- инсталляционный файл коммуникационного сервера *Visor\_on\_setup.exe*;
- инсталляционный файл клиента-просмотрщика *Visor\_setup.exe*;
- драйвер *Visor\_USBkey.inf* для ключа аппаратной защиты программного обеспечения “Visor-Key”;
- программный блок **FC10**, написанный на языке Step7, который используется для заполнения циклического буфера в режиме “SYNCHRONOUS” (номер блока можно изменять).

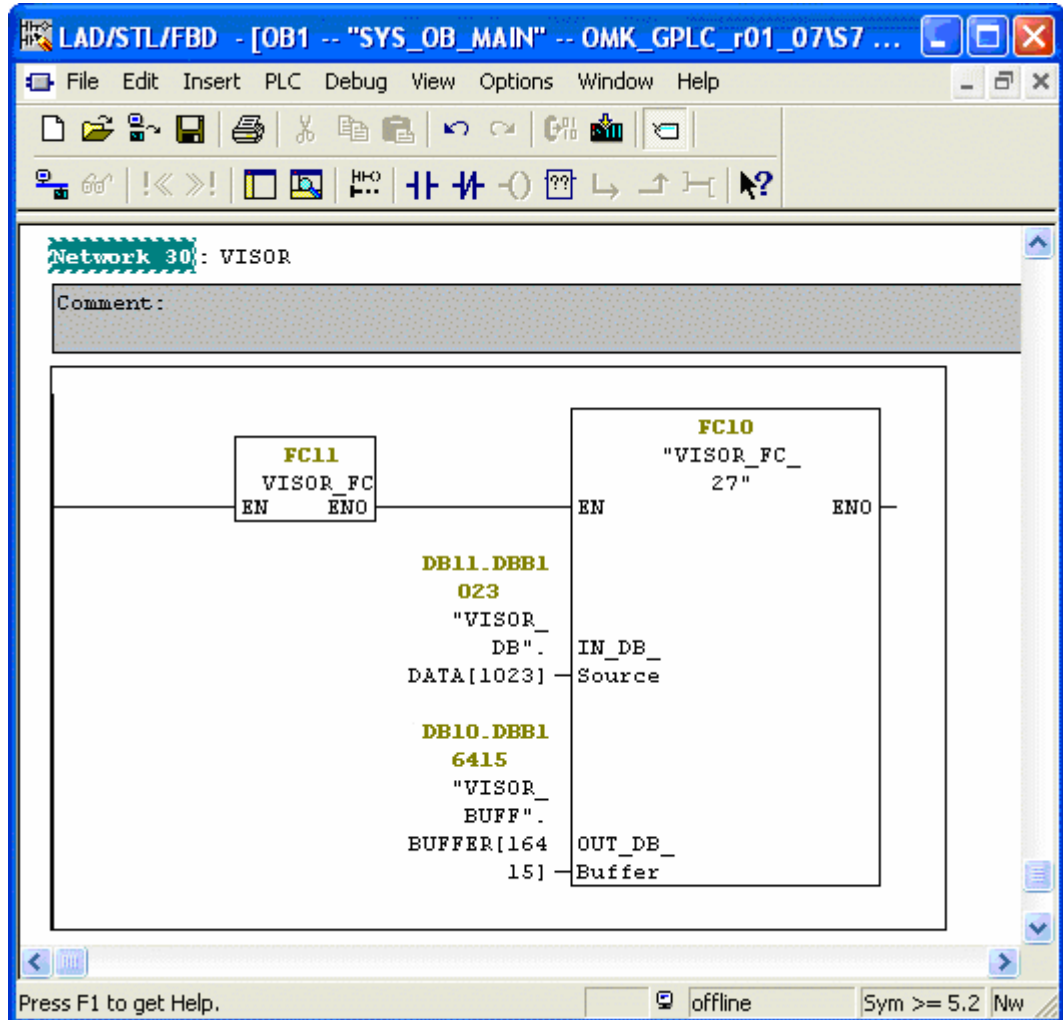
### 4.2. ПОСЛЕДОВАТЕЛЬНОСТЬ УСТАНОВКИ

Последовательность инсталляции программного обеспечения приведена ниже.

1. Запустить *Visor\_on\_setup.exe* и установить приложение коммуникационного сервера. В окне процедуры инсталляции необходимо указать путь установки.
2. Запустить *Visor\_setup.exe* и установить приложение клиента-просмотрщика. Если предполагается его совместное использование с коммуникационным сервером, то в окне процедуры инсталляции необходимо указать тот же путь установки.
3. Если списки и группы тегов уже имеются, то следует разместить их в папках “DBV” и “GRV” внутри рабочей директории “VISOR”.
4. Подсоединить ключ к любому свободному порту USB станции и установить по запросу драйвер ключа аппаратной защиты, используя поставляемый в комплекте файл *Visor\_USBkey.inf*.
5. Запустить коммуникационный сервер *Visor\_on.exe* и сконфигурировать параметры каналов в соответствии с проектом, не активируя их.
6. Во всех PLC создать блок данных (например, DB11) и обеспечить его заполнение необходимыми переменными процесса. Если используется режим “SYNCHRONOUS”, то в PLC дополнительно нужно создать блок данных циклического буфера (например, DB10) и запрограммировать вызов функционального блока FC10 из цикла программы. Номера блоков могут быть произвольными. На Рисунке 4-1 показан пример такой организации в PLC. Блок FC11 заполняет данными блок данных DB11. Первый входной параметр блока FC10 “IN\_DB\_Source” указывает на последний читаемый байт исходного блока данных, второй параметр “OUT\_DB\_Buffer” - на последний байт блока данных циклического буфера.
7. Настроить синхронизацию и время процессоров PLC, используя стандартные средства программирования.
8. Активировать необходимые каналы.

**Внимание!** Коммуникационный сервер **Visor\_On** не запустится в случае отсутствия ключа аппаратной защиты. Также не следует во время работы приложения вынимать ключ защиты из порта USB рабочей станции. Отсутствие ключа влечет за собой аварийную работу приложения

**Visor\_On** и появление ошибок в базе хронологических данных.



**Рисунок 4-1.** Пример вызова функционального блока обслуживания циклического буфера



## 5. USB КЛЮЧ АППАРАТНОЙ ЗАЩИТЫ VISOR-KEY

Для защиты от несанкционированного использования программное обеспечение “**VISOR**” поставляется с USB ключом **Visor-Key**. Защищен только коммуникационный сервер **Visor\_On**, клиент-просмотрщик **Visor** может использоваться без ограничений.

Ключ содержит одну лицензию на 1, 5, 10, 20 или 30 каналов. В случае с демонстрационной версией, ключ содержит ограниченную по времени лицензию, которая в дальнейшем может быть заменена на неограниченную. Процедура замены лицензии встроена в приложение **Visor\_On**. Файл лицензии может быть получен по электронной почте. Информация о лицензии находится в окне “*About*” приложения **Visor\_On**.

## 6. ПОДДЕРЖКА И КОНТАКТЫ

Получить последние версии программного обеспечения, описания продуктов, цены и другую информацию, вы можете на нашем сайте: <http://visor.su>

Любые замечания, комментарии, предложения и вопросы вы можете послать нам на электронную почту: [visor@visor.su](mailto:visor@visor.su)

Контактная информация:

- Тел.: 8-913-330-6755
- Адрес: 654038, г. Новокузнецк, пр-кт Советской Армии, д. 54, офис 52
- E-mail: [visor@visor.su](mailto:visor@visor.su)
- www: <http://visor.su>

---

© 2010 ООО "Визор"  
Все права защищены.