

**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**  
**"АРМ LanMon 3.6"**

Ревизия 7 от 30.08.2007 г.

**«МНПП Сатурн»**  
[www.mnppsatur.ru](http://www.mnppsatur.ru)

## Содержание

Содержание.....	2
Используемые термины.....	3
Общее описание системы LanMon.....	4
Драйверы оборудования.....	7
Просмотр списка каналов, экспортируемых драйвером.....	7
Просмотр статистики работы драйвера.....	8
Управление драйверами.....	9
Программная поддержка.....	9
Каналы.....	11
Формулы каналов.....	13
Группы каналов.....	14
Связи.....	17
Дискретные алармы.....	19
Аналоговые алармы.....	23
Методика создания аналогового аларма.....	25
Программы на бейсике.....	27
Программы на скриптовых языках.....	29
Доступ к базам данных.....	31
Тренды.....	32
Автопилоты.....	37
Отчеты RTF.....	39
Генератор отчетов (новый).....	41
Библиотеки картинок.....	42
Наборы картинок.....	44
Карты.....	46
Отображаемые объекты карты.....	50
Динамический объект на бейсике.....	53
Динамический объект на скриптовых языках.....	58
ActiveX объекты.....	64
Меню карты.....	68
Список отображаемых объектов на картах.....	69
IP телефония по протоколу H.323.....	71
Формат комплексного номера переговорного устройства сети СОС-95 (БГС, УИР-Р, БДКЛ).....	77
Особенности разговора с устройствами СОС-95.....	77
Как настроить микшер.....	78
Запись переговоров.....	80
Программная поддержка.....	80
Режимы APM LanMon.....	82
Настройки проекта.....	83
Журнал событий.....	91
Просмотр архива.....	92
Редактор выборки.....	94
Модуль GSM.....	95
Требования к аппаратному и программному обеспечению.....	96

## Используемые термины

**Канал** - переменная определенного типа. Несет информацию об одном аналоговом или дискретном параметре. Имеет уникальный адрес, метку времени, «качество» и значение. Канал это основной тип данных системы LanMon. Обычно канал привязывается к реальному датчику на объекте. Значение канала несет показание датчика. Качество канала несет информацию об исправности датчика и контроллера, к которому этот датчик подключен. Канал очень похож по смыслу на тэг OPC (технология *Ole For Process Control*).

**Адрес канала** – 4 числа. Каждое число адресует соответствующий уровень дерева каналов. Допускаются значения чисел адреса в диапазоне 1...65535. Адрес канала присваивается при наладке системы и никогда не меняется.

**Качество канала** – то же, что и «состояние канала» - поле STATE. Расшифровка поля STATE приведен в следующей таблице.

Значение	Расшифровка
0	ОК
1	Выключен (т.е. сознательно не опрашивается)
2	Состояние не определено (Нет никаких данных)
3	Неисправно само устройство (интеллектуальный датчик)
4	Неисправен контроллер
5	Значение недостоверно (что-то с датчиком)
6	Не подключен (проблема с каналом связи или с управляющей программой)

**Значение канала** – значение, которое несет канал. Зависит от типа канала. Тип данных значения расшифровывается полем DTYPE. Значение достоверно только если качество канала – «ОК» (т.е. поле STATE=0).

**Дерево каналов** – для группировки каналов они содержатся в 4х уровневом дереве.

**Сообщение** – изменение одного из параметров канала: значения, качества или метки времени. Сообщение может поступать от драйвера оборудования или от сервера LanMon. При поступлении сообщение регистрируется в журнале и в архиве APM LanMon.

**APM LanMon** – склада-система. Программа по обработке и отображению состояния промышленного оборудования. Имеет встроенные языки программирования. Может работать как с сервером LanMon так и без него. Данное руководство именно про эту программу.

**Драйвер оборудования LanMon** – DLL библиотека для взаимодействия APM LanMon с оборудованием.

**Сервер LanMon** – это отдельное приложение. Взаимодействует с клиентами (**APM LanMon**) и опросчиками по протоколу TCP/IP. Получает параметры от опросчиков и отправляет всем клиентам в режиме реального времени. Используется для организации работы системы LanMon в сетях TCP/IP.

**Клиент LanMon** - узел, подключенный к серверу LanMon по TCP/IP и постоянно получающий изменения определенной группы каналов в реальном времени. Типовой клиент это APM LanMon.

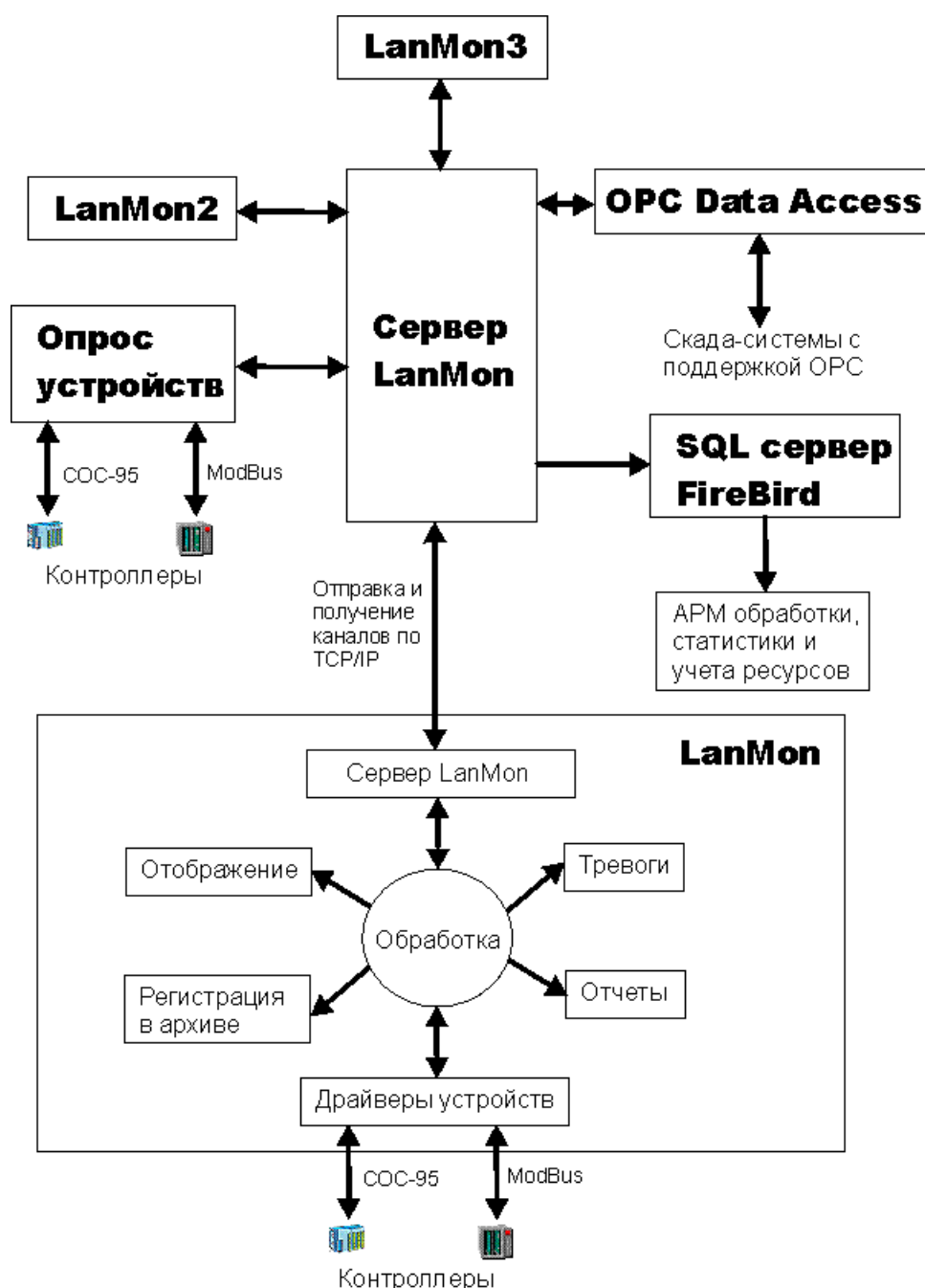
**Опросчик LanMon** – программа, подключенная к серверу LanMon по TCP/IP и записывающая изменения определенной группы каналов на сервер по мере их изменения. Чаще всего это отдельная программа для взаимодействия с оборудованием. Это может быть и контроллер с поддержкой протокола TCP/IP.

## Общее описание системы LanMon

LanMon это комплекс программ для обработки и отображения состояния промышленного оборудования. Система LanMon состоит из следующих основных программ:

- **АРМ LanMon** – основная программа комплекса. Имеет встроенную систему драйверов для взаимодействия с оборудованием. Имеет 5 встроенных языков программирования. Может работать как с сервером LanMon, так и без него. Для простых проектов с одним рабочим местом сервер не требуется. Данное руководство описывает эту программу.
- **Драйвер LanMon** – Драйвер для взаимодействия с оборудованием. Представляет собой DLL библиотеку, которая экспортирует предопределенный набор функций. Драйвер работает в среде АРМ LanMon. Драйвера бывают разные, например: bkd.dll – драйвер для контроллера системы СОС-95 БКД, modbus.dll – драйвер протокола ModBus и т.д.
- **Сервер LanMon** – Сервер каналов реального времени. Взаимодействует с программой АРМ LanMon и опросчиками по протоколу TCP/IP. Используется для организации сетевой информационной системы с несколькими рабочими местами и/или точками подключения к оборудованию.
- **Опросчик LanMon** – программа, подключенная к серверу LanMon по TCP/IP и записывающая изменения определенной группы каналов на сервер по мере их изменения. Чаще всего это специальная программа для взаимодействия с оборудованием. Например: программа OPROS2, которая работает со всем спектром оборудования и контроллеров системы СОС-95. А также, программа Lift4 – АРМ диспетчера лифтовой системы СЛДКС. Lift4 может работать опросчиком для сервера LanMon. Существуют контроллеры, которые являются опросчиками LanMon.
- **LanMon to OPC Data Access 2.0** – сервер OPC DA. Данная программа подключается к серверу LanMon по TCP/IP, получает от него определенную группу каналов и преобразует их в тэги OPC. Клиенты, поддерживающие OPC DA 2.0, подключаются к этой программе для получения тэгов в режиме реального времени. Используется для интеграции LanMon с другими склада-системами с поддержкой OPC DA.

Все программы системы LanMon работают под управлением операционных систем Windows 98/ME/NT 4.0/2000/XP. Настоятельно рекомендуется использовать Windows платформы NT: NT 4.0/2000/XP и файловую систему NTFS. На следующем рисунке представлена структурная схема распределенной системы на базе LanMon:



В нижней части рисунка показана схема прохождения данных (каналов) в программе АРМ LanMon. Драйверы устройств получают информацию о состоянии оборудования из контроллеров и преобразуют ее в каналы LanMon. Каналы поступают в модуль обработки. Модуль обработки АРМ LanMon может записывать данные в контроллеры через драйверы устройств. От модуля обработки каналы поступают в модуль графического отображения, модуль регистрации в архиве, модуль тревог, модуль отчетов. Модуль обработки может получать и отсылать каналы на сервер LanMon.

Возможна такая схема работы: каналы, получаемые от драйверов, обрабатываются и отсылаются на сервер, при этом, от сервера также получают и обрабатываются определенные каналы. В таком режиме сервера программа АРМ LanMon является одновременно опросчиком и клиентом. Если нам нужно одно рабочее место и есть один источник данных - сервер LanMon не используется.

«Опрос устройств» на рисунке – это программа, являющаяся опросчиком LanMon.

«LanMon №2» и «LanMon №3» - это автоматизированные рабочие места (АРМ) на базе программы АРМ LanMon, которые получают все каналы только от сервера. При этом на них


могут быть возложены совершенно разные задачи. Например: на рабочее место АРМ LanMon выводится охранно-пожарная, газовая сигнализации, контроль и управление силовым оборудованием. На АРМ «LanMon №2» выводится только охранно-пожарная сигнализация объекта и он установлен в комнате охраны, а на АРМ «LanMon №3» выводится только контроль и управление силовым оборудованием объекта.

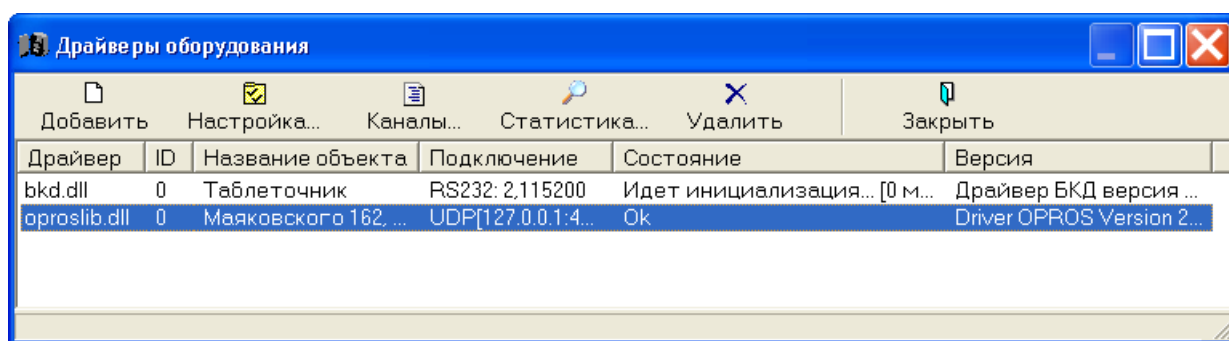
«OPC Data Access» - позволяет открыть доступ к определенной группе параметров на чтение и/или запись по протоколу OPC DA.

Сервер LanMon может работать в режиме постоянного архивирования всех каналов в SQL сервер. С SQL сервером могут работать рабочие места по статистике и учету ресурсов промышленного оборудования.

## Драйверы оборудования

APM LanMon поддерживает механизм драйверов оборудования. Драйвер представляет собой динамически линкуемую библиотеку (dll), которая экспортирует определенный набор функций. Драйвер отдает в APM LanMon поток значений каналов по аналогии с сервером LanMon. Драйверы должны находиться в директории проекта «..\DRIVER\». В комплекте стандартной поставки APM LanMon имеет драйвер контроллеров семейства БКД (bkd.dll). Дополнительно поставляется драйвер приборов теплоучета (oproslib.dll) с поддержкой записи архивных данных в SQL базу данных.

Для просмотра списка драйверов проекта выберите пункт «Ресурсы\Драйверы оборудования...» главного меню или нажмите кнопку  в панели главного окна. Появится окно следующего вида:



Приведем пояснение значений колонок списка драйверов:

- Драйвер – название DLL библиотеки драйвера.
- ID – номер экземпляра данного драйвера в списке с нуля. Один драйвер может быть использован несколько раз, например: надо работать с двумя однотипными контроллерами, подключенными к разным портам RS-232 компьютера;
- Название объекта – название объекта, который обслуживает данный драйвер.
- Подключение – тип подключения к аппаратуре, который реализует данный драйвер.
- Состояние – текущий статус работы драйвера. Эта колонка обновляется автоматически 1 раз в секунду.
- Версия – описание версии данного драйвера.

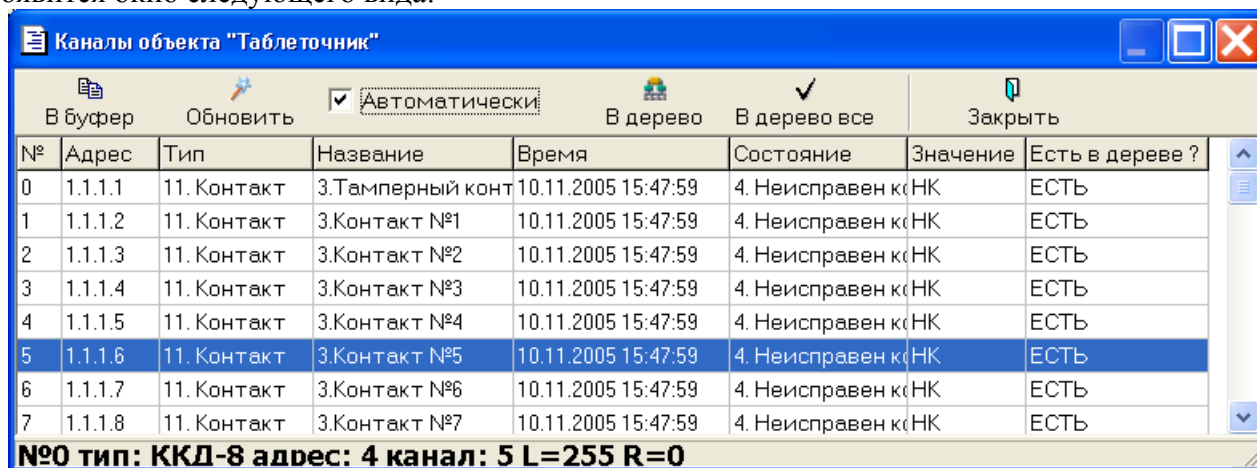
Кнопка «Добавить...» служит для добавления нового драйвера в список используемых в APM LanMon, вам надо лишь указать файл с DLL библиотекой драйвера. Кнопка «Удалить» удаляет драйвер из списка используемых в APM LanMon. Кнопка «Настройка...» вызывает окно настройки параметров работы драйвера. Окно настроек является специфичным для каждого драйвера и описано в документации на драйвер.

После добавления драйвера в список и его настройки он сразу начинает работать (осуществлять опрос оборудования и поставлять информацию в APM LanMon). Перезапускать APM для этого не требуется. Получая данные от драйверов оборудования, APM LanMon может отправлять их на сервер LanMon. Для включения этого режима в настройках проекта установите режим работы с сервером «Опрос».

Если планируется, что APM LanMon получает данные только от сервера – список драйверов должен быть пуст.

### Просмотр списка каналов, экспортируемых драйвером

Для просмотра списка каналов, поставляемых драйвером, нажмите кнопку «Каналы». Появится окно следующего вида:



Приведем пояснение значений колонок списка драйверов:

- **№** - порядковый номер канала в списке;
- **Адрес** – адрес канала;
- **Тип** – тип канала и его номер;
- **Название** – название канала;
- **Время** – время последнего изменения значения канала;
- **Состояние** – состояние канала и его номер;
- **Значение** - значение канала;
- **Есть в дереве** - признак наличия канала с таким адресом в списке каналов APM LanMon (в дереве каналов);

Кнопка «В буфер» копирует содержимое списка каналов буфер обмена Windows.

Кнопка «Обновить» однократно обновляет список каналов.

Галочка «Автоматически» включает режим обновления списка каналов каждую секунду.

Кнопка «В дерево» добавляет в дерево каналы, которых там еще нет.

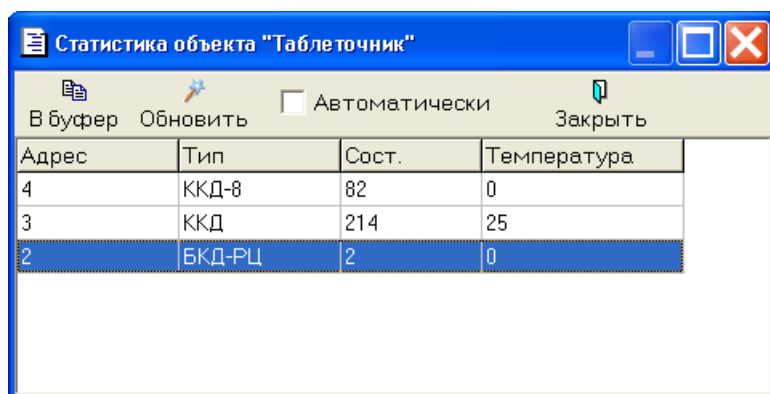
Кнопка «В дерево все» добавляет в дерево все каналы. Если такой канал уже есть в дереве – он будет перезаписан.

Для разработки нового драйвера доступен комплект разработчика. Он включает в себя пример исходного текста динамической библиотеки драйвера на языке C++ и описание интерфейса. Комплект разработчика распространяется бесплатно.

### Просмотр статистики работы драйвера

Для просмотра статистики по работе оборудования, опрашиваемого драйвером, нажмите кнопку «Статистика». Появится окно следующего вида:






Кнопка «Обновить» однократно обновляет список каналов.

Галочка «Автоматически» включает режим обновления (запроса у драйвера) статистики каждую секунду.


Название и содержание колонок статистики определяет сам драйвер. У разных драйверов вид и содержание статистики различается.

### Управление драйверами

Если в списке драйверов есть хоть один драйвер, APM LanMon создает в панели задач значок . Если подвести курсор мыши к значку – возникает всплывающая подсказка, в которой указано общее количество драйверов и количество драйверов со статусом «ОК» (подключен к оборудованию, нормально работает). Если количество драйверов со статусом «ОК» меньше общего количества драйверов – значок в панели задач становится перечеркнутым.

Двойной щелчок мышью на значке вызывает окно драйверов. При нажатии правой кнопки мыши на значке возникает контекстное меню с пунктом «Завершение работы». Этот пункт завершает работу APM LanMon. В параметрах проекта на вкладке «Драйверы» есть опция, позволяющая запретить завершение работы из контекстного меню. В режиме исполнения проекта все работает также.

Например, нам нужно использовать APM LanMon в режиме «Опрос» без интерфейса пользователя. Для этого надо:

- Настроить необходимые драйверы оборудования;
- В параметрах проекта выбрать режим работы с сервером LanMon «Опрос»;
- В параметрах проекта на вкладке «Карты» очистить список карт для автоматической загрузки;
- В параметрах проекта на вкладке «Драйверы» установить галочку «Разрешить завершение работы программы из контекстного меню»;
- Запустить проект на выполнение. При этом в панели задач появится значок  для доступа к списку драйверов.

### Программная поддержка

Для доступа к статистике по работе драйверов во время выполнения проекта, в скрипте есть две функции:

- DriverShow(Show: Boolean) - Показать или скрыть окно драйверов оборудования. В режиме выполнения проекта функции настройки, добавления и удаления каналов недоступны.

- `DriverChannelInfo(A1,A2,A3,A4: Word; Delim: Char): String` - Получить дополнительную информацию о физическом подключении указанного канала от драйверов оборудования. В качестве входного параметра надо задать адрес канала. Информация возвращается в виде текстовой строки. Можно задать разделитель для полей (по умолчанию это символ табуляции). Если драйверы оборудования не используются, функция возвращает пустую строку.

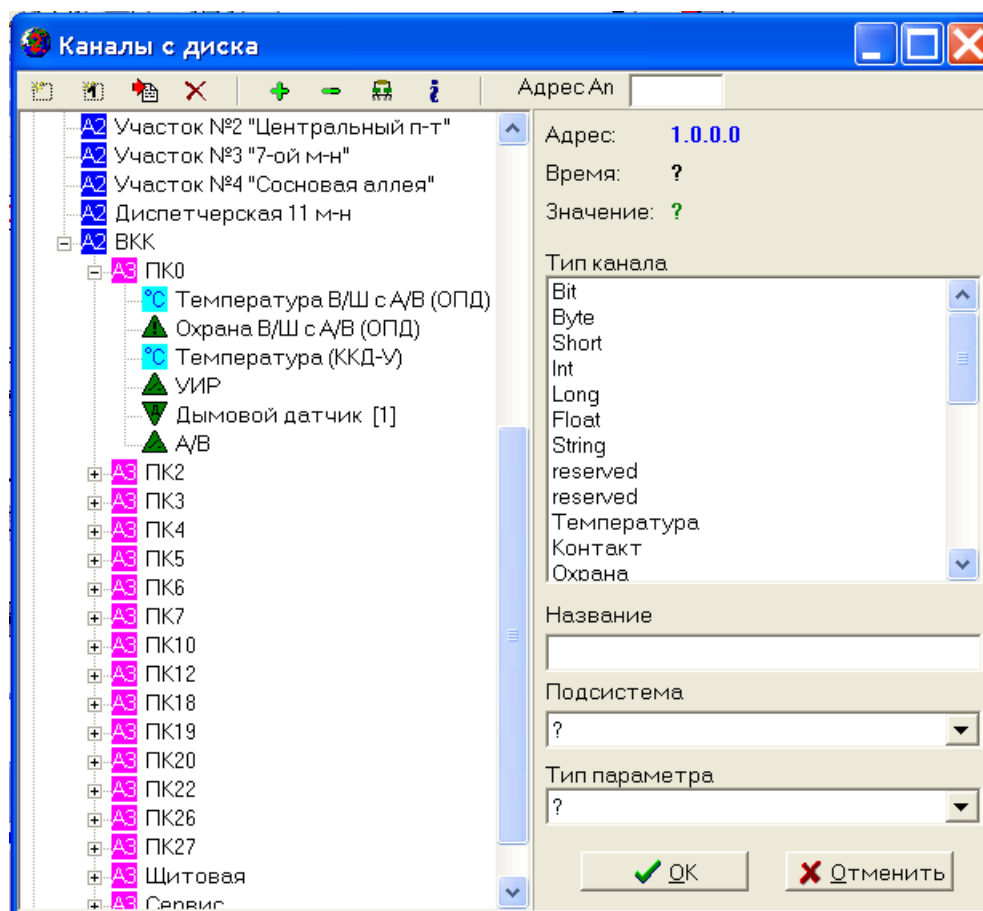
Типовое применение функции `DriverShow`: отдельная кнопка вызывает данную функцию для показа статистики по работе драйверов.

Типовое применение функции `DriverChannelInfo`: из контекстного меню объекта карты вызывается эта функция для получения информации о физическом подключении канала данного объекта карты.

Более подробное описание этих функций содержится в «Руководстве по программированию».

## Каналы

Каналы APM LanMon представляются в виде 4-х уровневого дерева. Для вызова редактора каналов выберите пункт главного меню «Ресурсы/Каналы...». Появится окно следующего вида:



В верхней части окна расположена панель инструментов для редактирования дерева каналов. Приведем назначение ее кнопок:

- Добавить канал на уровень ниже текущего;
- Добавить канал на тот же уровень, что и текущий;
- Изменить параметры канала;
- Удалить текущий канал или целую ветвь;
- Раскрыть все дерево каналов;
- Собрать все дерево;
- Раскрыть текущую ветвь;
- Получить статистику по каналам в текущей ветви дерева;

В правой части окна расположены параметры канала: тип канала, название, подсистема, единицы измерения. Перед добавлением нового канала или для изменения существующего надо выставить параметры канала.

В заголовке окна указано: откуда получено и где хранится данное дерево каналов:

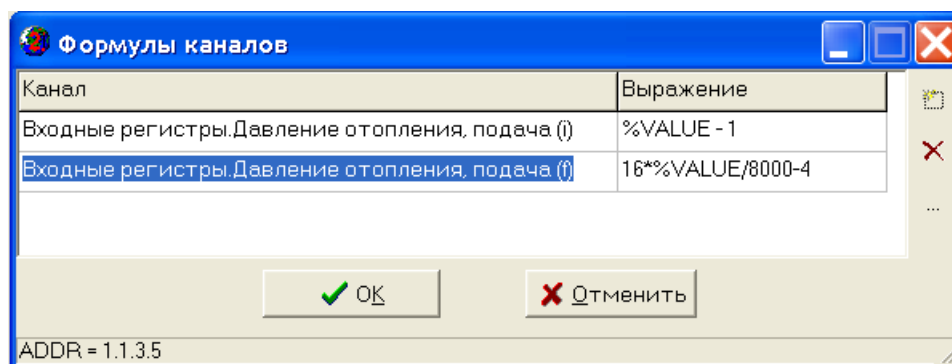
- Каналы с диска – редактор работает с каналами из файлов «..\LanMon\\*.dat». Данный режим активен, когда не используется подключение к серверу или в настройках программы снята галочка «Получать каналы с сервера».

- Каналы с сервера – редактор работает с каналами на сервере. Данный режим активен, когда используется подключение к серверу и в настройках программы установлена галочка «Получать каналы с сервера».

## Формулы каналов

Формула канала это выражение для пересчета значения канала, полученного от драйвера оборудования или от сервера. Например, канал с адресом 1.1.3.5 имеет тип float и содержит код АЦП с датчика давления. Для обработки в АРМ LanMon (отображения на экране, регистрации в архиве и пр.) необходимо представить давление в Бар. Надо произвести пересчет закодированного значения напряжения АЦП в единицы давления в соответствии с характеристикой датчика давления. Выведем формулу пересчета. Значение кода АЦП в выражение заносится как %VALUE (это аргумент функции). Выражение может использовать все операторы и функции встроенного языка Бейсик. Выражение вычисляется после изменения состояния канала с качеством «ОК» один раз.

После получения формулы пересчета добавим ее в формулы каналов. Для этого вызовите окно «Формулы каналов» из меню «Ресурсы/Формулы каналов...». Появится окно следующего вида:



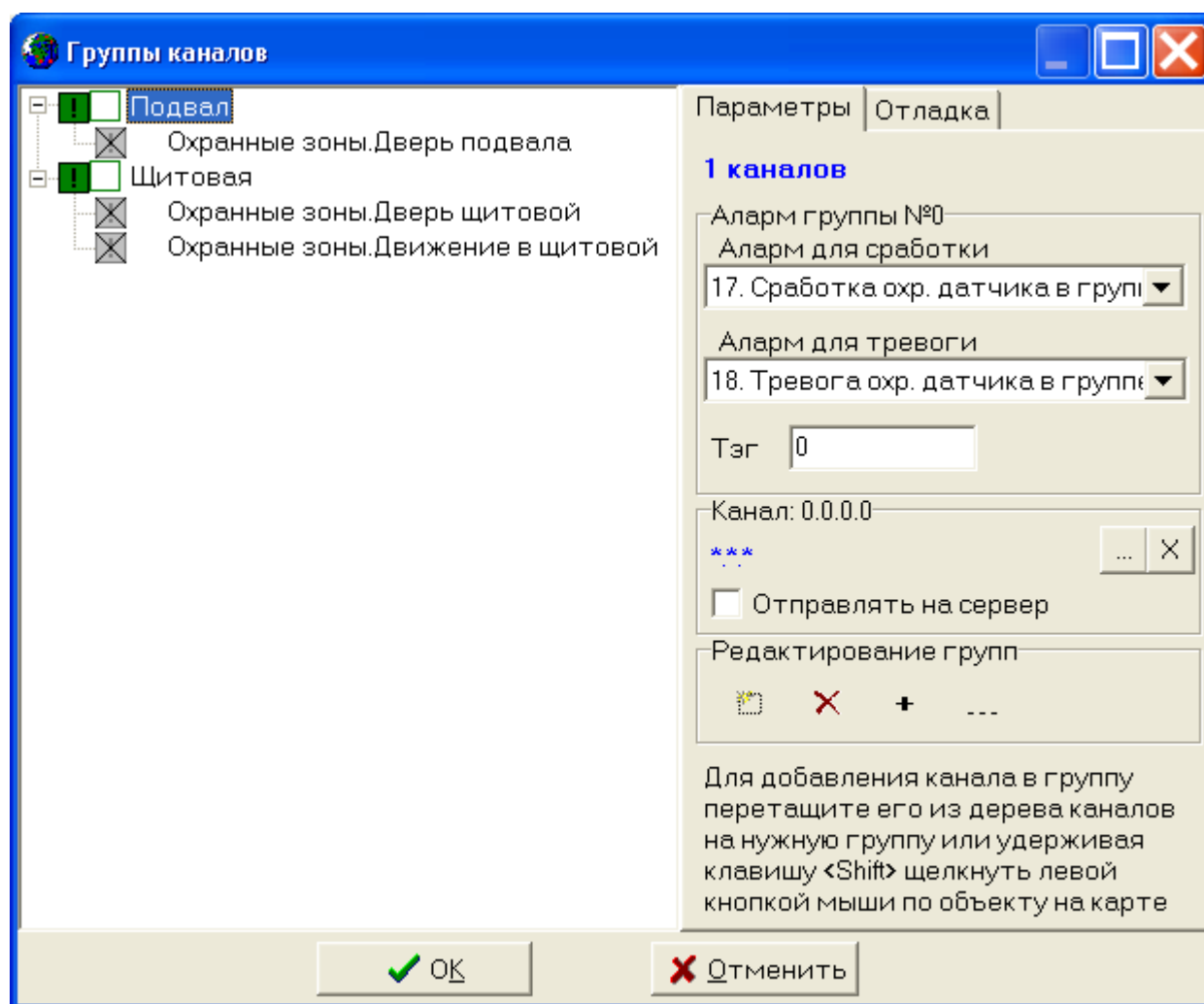
Для добавления формулы канала нажмите кнопку . Введите в поле «Выражение» формулу пересчета. Для ввода адреса канала нажмите кнопку или дважды щелкните мышью в поле «Канал» или нажмите клавишу «Enter» - появится окно выбора канала. Выберите канал и нажмите клавишу «ОК». Для удаления формулы канала из списка нажмите кнопку . Для сохранения изменений нажмите клавишу «ОК». Новая формула начинает действовать сразу на вновь приходящие события.

## Группы каналов

Несколько каналов могут быть объединены в группу. В группу могут входить как конечные каналы, так и «маски каналов». Например, если в группу добавлен канал с адресом 49.12.1.0, это означает что все имеющиеся в конфигурации каналы с адресом 49.12.1.\* (вместо «\*» - любое число) входят в группу. Группы каналов создаются один раз на стадии пуско-наладки конкретного проекта.




Группы каналов применяются, например, для создания охранных зон. Каналы от охранных датчиков объединяются в группу (охрану зону). Это позволяет просто управлять постановкой на охрану этих каналов и обрабатывать события от них указанным алармом.




Для редактирования групп каналов выберите пункт главного меню «Ресурсы/Группы каналов...». Появится окно следующего вида:



В левой части окна групп каналов расположено дерево групп и входящих и их состав каналов. При изменении выделения в дереве групп, в правой верхней части окна расшифровывается состояние группы или канала. Состояние элемента дерева отображается дополнительно значками в дереве:

- группа снята с охраны
- группа на охране
- группа или канал в норме
- группа или канал в состоянии срабатывания (определяется алармом для сработки)

-  - группа или канал в состоянии тревоги (определяется алармом для тревоги)
-  - группа или канал в неисправности
-  - состояние канала «выключен» или «нет данных»

Для добавления новой группы каналов нажмите кнопку . Для удаления группы или отдельного канала из группы нажмите кнопку . Чтобы распаковать или собрать дерево групп нажмите кнопку . Для изменения названия группы наведите на него курсор и щелкните левой клавишей мыши или нажмите кнопку «F2». Для добавления канала в группу окно с деревом каналов из меню «Ресурсы/Каналы...» и с использованием мыши «перетащите» канал из дерева каналов в дерево групп.

Каждой группе каналов можно назначить два дискретных аларма и тэг. Цифровые алармы используются для формирования состояний «Сработка» и «Тревога» соответственно (см. [Дискретные алармы](#)). Тэг представляет собой дополнительный параметр – число целого типа, которое можно привязать к группе каналов. Это число доступно из программы по свойству Tag. Поле тэг может использоваться произвольно по усмотрению программиста.

Группе каналов можно назначить канал с типом 21 «Состояние локальной охранной зоны». Группа сама будет формировать значение этого канала. Вместе с изменением состояния группы, он будет принимать следующие значения:

- 0-охрана снята*
- 1-на охране*
- 2-снят с охраны, срабатывание*
- 3-на охране, срабатывание*
- 4-снят с охраны, тревога*
- 5-на охране, тревога*

Если установить галочку «Отправлять на сервер» - этот канал будет отправляться на сервер LanMon. Такой режим работы используется для синхронной постановки / снятия охраны групп между разными рабочими местами в сети.

В программе на скрипте группа каналов представлена классом TGroup. У нее есть следующие свойства:

- Name - название группы каналов, заданное в настройках группы
- Neisprav - признак наличия в группе хоть одного неисправного канала
- Ohrana - состояние охраны группы каналов
- State - состояние срабатывания хоть одного канала в группе.
- Tag - свойство типа int. Используется для хранения информации, привязанной к группе каналов.

Динамически создавать группы нельзя. Можно обращаться только к группам, определенным на этапе редактирования проекта.

Список групп представлен классом TGroups. По умолчанию определена глобальная переменная Groups (экземпляр класса TGroups). У TGroups есть свойства Count, Index[] для доступа к каждой группе из списка.

Например, установка охраны группы с номером *i* осуществляется командой: `Groups.Items[i].Ohrana=true`; Номер группы каналов считается с нуля. Состояние охраны можно считывать и устанавливать.

Состояние неисправности группы устанавливается когда один из ее каналов имеет статус «неисправен датчик», «неисправен контроллер», «значение недостоверно» или «не подключен». Это свойство доступно из программы только на чтение командой `Groups.Items[i].Neisprav` для группы номер *i*. Значение логическое: false – нет неисправности, true – неисправность.

Состояния нормы, срабатывания и тревоги в группе устанавливаются, когда сработал соответствующий аларм по одному из каналов, входящих в группу. Состояние тревоги более

приоритетно. Доступно из бейсика только на чтение командой *Groups.Items[i].State* для группы номер *i*. Значение - целое число: 0-норма 1-срабатывание 2-тревога.

Пример программы для работы с группами Вы можете найти в файле «..\program\samples\LanMon\c++\groups.cpp».

Состояние охраны группы используется для организации охранных зон. Каналы от охранных извещателей объединяются в группы. На одной карте могут размещаться каналы из нескольких охранных зон. Установка и снятие охраны группы осуществляется из программы на скрипте.

На вкладке «Отладка» расположены следующие кнопки:

- «Поставить на охрану» - ставит на охрану текущую группу;
- «Снять с охраны» - снимает с охраны текущую группу;
- «Состояние групп» - отобразить в дереве групп текущие состояния всех групп и каналов;

Можно запустить проект на выполнение, оставив открытым окно редактора групп каналов. Таким образом, можно просматривать и управлять состоянием групп в режиме выполнения проекта.

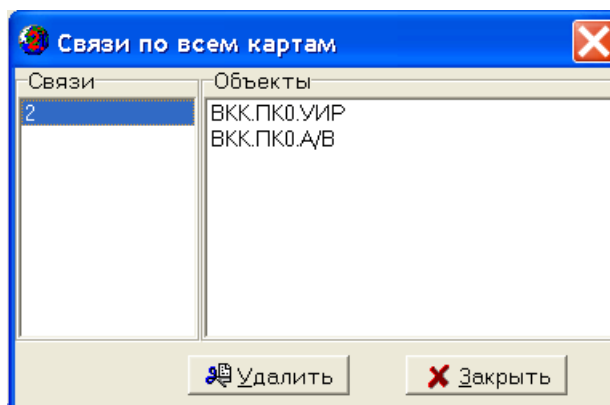
Пример работы с группами каналов приведен в демонстрационном проекте «Пример проекта LanMon C++». Этот проект устанавливается в комплекте с APM LanMon.



## Связи

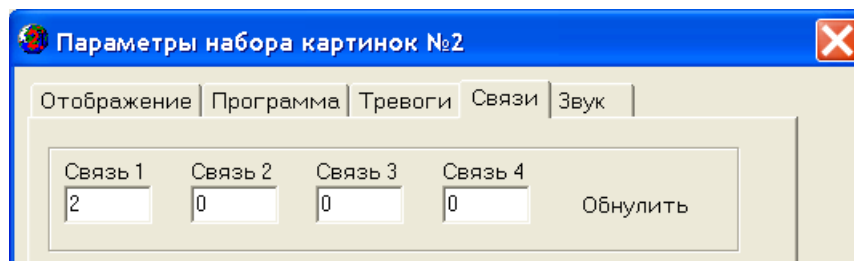
Связь – это логическое объединение двух и более каналов. Связи каналов нужны для фильтрации ложных сработок датчиков. Алгоритм работы связей заключается в следующем. При срабатывании канала АРМ LanMon смотрит: участвует этот датчик хоть в одной связи или нет. Если нет – срабатывание сразу получает статус тревоги. Если да, то АРМ LanMon смотрит: было ли срабатывание связанного со сработавшим датчика менее, чем «Задержка мин.» назад. Если да, то оба срабатывания идут как тревожные. Если нет, то последнее срабатывание не считается тревогой. Если хоть один из каналов, входящих в состав связи, неисправен или замаскирован - все работает, как будто данной связи нет вообще. Надо помнить, что механизм связей работает только если установлена галочка «Использовать связи» в настройках АРМ LanMon. Если эта галочка снята – все работает, как будто связей нет вообще. Все настройки для механизма связей можно сделать в настройках АРМ LanMon на вкладке «Тревога».

Для вызова окна просмотра имеющихся связей выберите пункт главного меню «Ресурсы/Связи...», появится окно следующего вида:



В левой части окна расположен список связей. Связи идентифицируются по номерам (с единицы). При выделении связи в списке в правой части окна показываются входящие в состав связи каналы. В составе связи должно быть не менее двух каналов. Кнопка «Удалить» позволяет удалить выделенную связь.

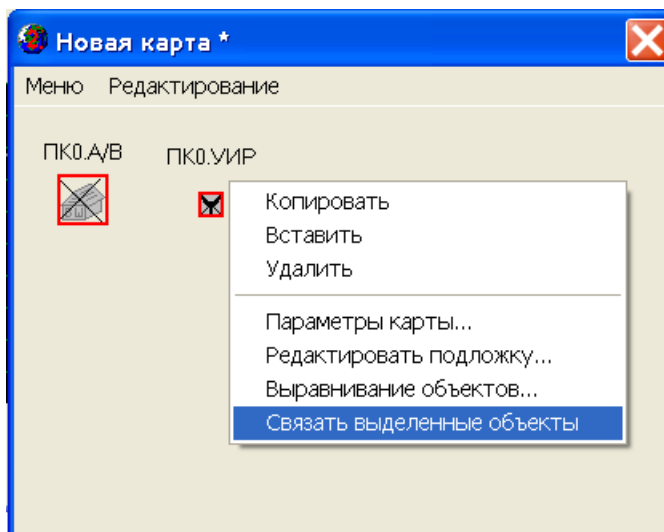
Связи хранятся в картах. Список связей показывает связи по всем картам, загруженным в данный момент. В связь может входить только канал, отображающийся на карте АРМ LanMon. Каждый канал может входить максимум в 4 связи. Для редактирования связи вызовите окно свойств отображаемого на карте объекта и перейдите на вкладку «Связи». Это окно выглядит так:



Номера связей, в которые входит данный объект надо вводить вручную. Для исключения канала из всех связей нажмите кнопку «Обнулить». Для добавления канала в связь номер N в свободное поле одной из связей вместо нуля введите число N.

Есть альтернативный способ создания связи. На карте АРМ LanMon, в режиме редактирования выделите объекты, которые надо объединить с созданием новой связи.

Выделение объектов производится левой клавишей мыши с удержанием левой клавиши «Ctrl». После выделения объектов щелкните правой клавишей мыши на карте и в появившемся контекстном меню выберите пункт «Связать выделенные объекты». Произойдет создание новой связи с первым свободным номером. Пример связывания объектов карты приведен на следующем рисунке.



Пункт меню карты «Редактирование/Сдвинуть нумерацию связей» позволяет прибавить или вычесть константу из всех связей на данной карте. Это используется, когда номера связей на разных картах пересекаются.

В основном, связи используются для охранных каналов при построении охранной сигнализации. Часто требуется формировать тревожное извещение при срабатывании не одного, а двух извещателей, например, для отсеечения ложных срабатываний объемных извещателей.

## Дискретные алармы

Дискретные алармы предназначены для мониторинга значений дискретных параметров (каналов). Под дискретными параметрами понимаются ступенчато изменяющиеся величины: извещения о срабатывании охранных или пожарных датчиков, переключения исполнительных механизмов и т.п. При удовлетворении значения параметра указанным условиям производится звуковое и/или визуальное оповещение оператора.

Дискретный аларм имеет два состояния: норма и тревога. Описание каждого аларма состоит из:

- ☞ условия тревоги;
- ☞ текстовое сообщение о тревоге, сформатированное особым образом;
- ☞ звуковой файл, проигрываемый при возникновении тревоги;

АРМ LanMon имеет таблицу алармов, где каждый аларм имеет номер и название. Поддерживается до 1000 дискретных алармов. Алармы с 1 по 25 зарезервированы как стандартные для АРМ LanMon (это типовые реакции на срабатывания охранных и пожарных извещателей). После установки АРМ LanMon они уже настроены нужным образом. Остальные алармы (с 26 по 1000) могут использоваться произвольно. Настройки дискретных алармов хранятся в текстовом файле DALARM.INI в директории текущего проекта.

Для работы аларм должен быть привязан к отображаемому объекту карты. На карту надо добавить отображаемый объект, указать ему канал и указать ему заранее созданные алармы. Привязка алармов к объекту карты производится в окне свойств объекта на вкладке «Тревоги».

К каждому объекту на карте можно привязать до 3-х алармов. Например, к температурному каналу обычно привязываются 2 аларма: «Тревога по порогу температуры» и «Тревога по росту температуры».

Для настройки алармов выберите пункт меню «Ресурсы/Алармы...» и переключитесь на вкладку «Дискретные алармы». Появится окно следующего вида:

## Параметры аларма

«**Аларм разрешен**» - эта галочка определяет: будет аларм обрабатываться или нет.

«**Название**» - текстовая строка произвольного вида, поясняющая суть данного аларма.

Используется только на этапе редактирования проекта.

«**Условие тревоги**» - «**Всегда**» - тревога не зависит от состояния канала, «**Когда канал на охране**» - тревога может возникнуть только когда канал стоит на охране: группа каналов с этим каналом или карта стоят на охране.

«**№ значения**» - некоторые каналы содержат несколько значений. Значение номер ноль - это основное значение канала. Значение по умолчанию – ноль – используется в большинстве случаев.

### «Брать из»

- *Текущее качество* – проверяем текущее состояние канала;
- *Предыдущее качество* – проверяем предыдущее состояние канала;
- *Текущее значение* – проверяем текущее значение канала. Проверка осуществляется ТОЛЬКО если текущее качество «ОК», т.е. значение достоверно.
- *Предыдущее значение* – проверяем предыдущее значение канала. Проверка осуществляется ТОЛЬКО если предыдущее качество «ОК», т.е. значение достоверно.

«**Операция**» - действие, совершаемое над каналом для выяснения: аларм в норме или в тревоге.

☞ **нет** – операция не используется (ее результат всегда TRUE).

☞ **<** - меньше.

☞ **>** - больше.

- ☞ == - равно.
- ☞ != - не равно.
- ☞ <= - меньше или равно.
- ☞ >= - больше или равно.
- ☞ N1,N2..N20 – список возможных значений канала через запятую (максимум 20).
- ☞ UP Rate – скорость роста значения канала: тревога при росте «операнд/мин.»

Результаты операции 1 и операции 2 объединяются по логическому «И». Если результат равен TRUE – аларм формирует тревогу. Т.е. выдается окно тревог и производится звуковое оповещение оператора.

**«Сообщение»** - символьная строка, поясняющая суть произошедшей аварии, выдается пользователю APM LanMon в красном окне тревог. Если строка пустая – окно тревог в программе APM LanMon не выдается. Кроме обычных символов в данной строке могут присутствовать специальные подстановки, начинающиеся с символа %. При выдаче сообщения эти подстановки заменяются на:

- ☞ %DATE – дату события в формате ДД.ММ
- ☞ %TIME – время события в формате ЧЧ.ММ.СС
- ☞ %A1- название канала LanMon уровня A1
- ☞ %A2 - название канала LanMon уровня A2
- ☞ %A3 - название канала LanMon уровня A3
- ☞ %A4 - название канала LanMon уровня A4
- ☞ %STATE – название состояния данного канала
- ☞ %VALUEn(%mask) - значение канала. Где n – номер значения канала с нуля, %mask – маска для форматирования значения канала. Если маска форматирования опущена будет осуществлено форматирование по умолчанию. Маска форматирования значения описана в документе «Руководство по программированию» в приложении «Маска форматирования для функций Format и sprintf». Подстановка %VALUE выдает основное значение канала сформатированное по умолчанию.

**«Фильтрация выдачи сообщений в окно тревог, секунд»** - активизация аларма производится ТОЛЬКО если между тревожными значениями параметра прошло время большее указанного. Если задано значение ноль, то фильтрация использоваться не будет. Чаще всего используется для формирования извещений о срабатывании объемных охранных извещателей.

**«Звуковой файл»** - WAV файл, который надо проиграть при тревоге аларма. Если задано имя файла без пути, то файл будет искаться в поддиректории .\WAV\ текущего проекта. Если данное поле оставить пустым, то звуковое оповещение не производится.

**«Кол-во проигрываний звукового файла»** - сколько раз надо проиграть звуковой файл.

**«Пауза между проигрываниями, сек.»** - сколько секунд ждать между повторением проигрывания звукового файла.

**«Проиграть звуковой файл объекта»** - установите эту галочку, чтобы проиграть еще и файл, заданный на вкладке «Звук» в параметрах объекта.

**«Произносить номер пикета из названия канала»** - надо ли произносить номер пикета после проигрывания звукового файла, если название уровня A3 начинается с символов “ПКnnn”, где nnn номер от 1 до 999. Данная галочка используется только для коммуникационных коллекторов с размеченным пикетажем.

При срабатывании дискретного аларма, вызываются события в программе на скрипте: *OnDigitalAlarm* и *OnDigitalAlarm2*. Для добавления обработчиков этих событий в редакторе программ выберите в меню **«События / Сработка дискретного аларма тип 1»** или **«События / Сработка дискретного аларма тип 2»**. Вызов обработчиков производится, только если они определены в программе.

Обработчик на активизацию дискретного аларма тип 1 имеет следующие параметры:

*int Alarm* - номер дискретного аларма с 1.

*WORD A1, A2, A3, A4* - адрес канала, который вызвал активизацию аларма.

*bool & bAllow* – разрешение активизации аларма. По умолчанию значение **true**. Под активизацией аларма понимается появление окна алармов и включение проигрывания звукового оповещения.

Пример обработчика на активизацию дискретного аларма тип 1:

```
void OnDigitalAlarm(int Alarm, WORD A1, WORD A2, WORD A3, WORD A4, bool & bAllow)
{
    // Запретим активизацию аларма №27
    If( Alarm==27 ) bAllow=false;
    else bAllow=true;
}
```

Обработчик на активизацию дискретного аларма тип 2 имеет следующие параметры:

*int Alarm* - номер дискретного аларма с 1.

*TMap map* - ссылка на карту, где находится отображаемый объект, вызвавший активизацию аларма.

*TMonControl control* - ссылка на объект карты, вызвавший активизацию аларма.

*bool & bAllow* - разрешение активизации аларма (и появления окна алармов и звукового оповещения). По умолчанию значение **true**.

*bool & bVisual* - разрешение появления окна алармов. По умолчанию значение **true**.

*bool & bSound* - разрешение проигрывания звукового оповещения. По умолчанию значение **true**.

Пример обработчика на активизацию дискретного аларма тип 2:

```
void OnDigitalAlarm2(int Alarm,
    TMap map,
    TMonControl control,
    bool & bAllow,
    bool & bVisual,
    bool & bSound)
{
    // Запретим звуковое оповещение по активизацию аларма №27
    If( Alarm==27 ) bSound =false;
    else bSound =true;
}
```

## Аналоговые алармы

Аналоговые алармы предназначены для мониторинга значений аналоговых параметров. Под аналоговыми параметрами понимаются плавно изменяющиеся величины: показания датчиков давления, температуры, расхода и т.п. Для каждого канала определяются границы допустимого значения. При выходе значения за указанные пределы производится звуковое и/или визуальное оповещение оператора. АРМ LanMon имеет таблицу алармов, где каждый имеет номер и название. Всего может быть 1000 аналоговых алармов (номера с 1001 до 2000). Для редактирования алармов выберите пункт меню «Ресурсы/Алармы...» и переключитесь на вкладку «Аналоговые алармы». Появится окно следующего вида:

В левой части окна расположен список алармов. В правой – параметры аларма, выделенного в списке. Каждый аларм имеет следующие параметры:

«**Аларм разрешен**» - признак того надо обрабатывать аларм или нет.

«**Название**» - название аларма в списке. Используется только на этапе редактирования проекта.

«**Канал**» - адрес канала LanMon к которому привязан данный аларм.

«**Недопустимое повышение**», «**Повышение**», «**Мертвая зона**», «**Понижение**», «**Недопустимое понижение**» - допустимые границы значения канала. Их смысл объясняются ниже.

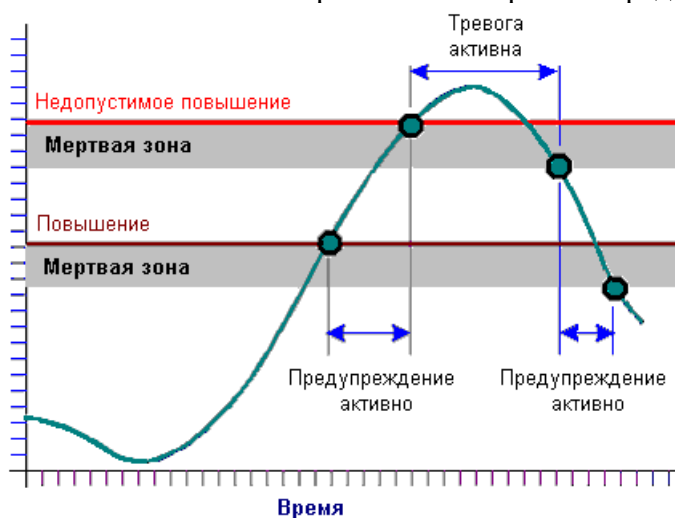
«**Сообщение**» - текстовое сообщение, которое выдается в окно аналоговых тревог при срабатывании аларма. В сообщение можно вставлять подстановки:

- ✓ %DATE – дата возникновения тревоги
- ✓ %TIME – время возникновения тревоги
- ✓ %A1 %A2 %A3 %A4 – элементы названия канала

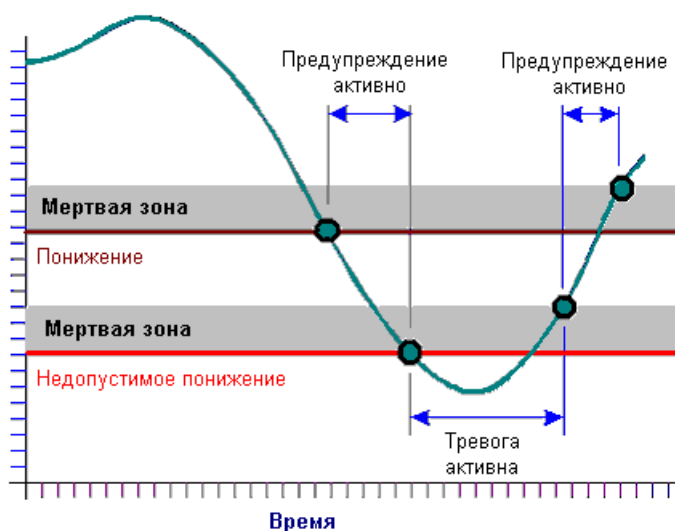
- ✓ %STATE – состояние канала (качество)
- ✓ %VALUEn(%mask) - значение канала. Где n – номер значения канала с нуля, %mask – маска для форматирования значения канала. Если маска форматирования опущена будет осуществлено форматирование по умолчанию. Маска форматирования значения описана в документе «Руководство по программированию» в приложении «Маска форматирования для функций Format и sprintf». Подстановка %VALUE выдает основное значение канала сформатированное по умолчанию.
- ✓ %ALARM – текущее состояние аларма (текстовая строка): "ОК", "повышение", "недопустимое повышение", "понижение", "недопустимое понижение".

Смысл параметров, касающихся звукового оповещения очевиден.

Логика работы аналогового аларма поясняется на следующих рисунках. Зеленым цветом показан график изменения значения канала. Превышение верхнего предела:



Пренижение нижнего предела:



На графиках обозначены участки:

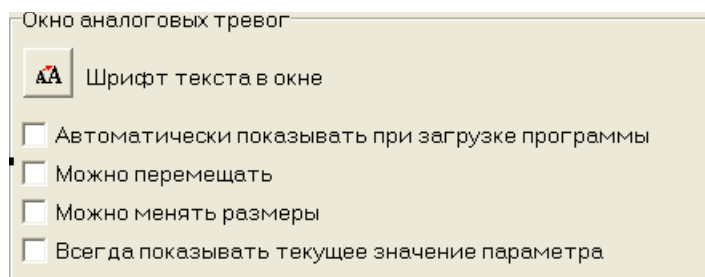
«**Тревога активна**» - в окно аналоговых тревог выдается сообщение красного цвета.

«**Предупреждение активно**» - в окно аналоговых тревог выдается сообщение желтого цвета.

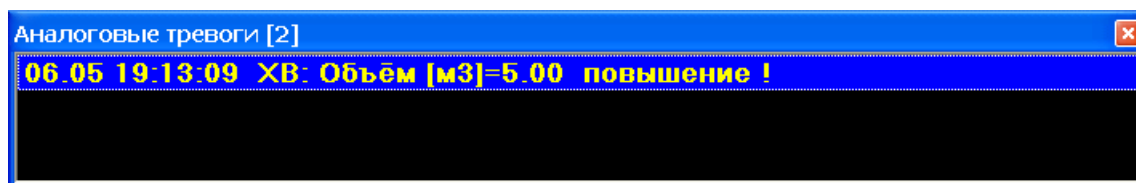
Звуковой файл проигрывается, если уровень аварийности аларма повышается: из нормального значения канала переходит в зону «Предупреждение активно» или из зоны «Предупреждение активно» в зону «Тревога активна».



Для отображения значений параметров и оповещения о выходе за установленные пределы используется окно аналоговых тревог. Для показа этого окна выберите в меню «Просмотр / Аналоговые тревоги...» в режиме редактирования проекта. В режиме выполнения проекта открыть окно аналоговых тревог можно вызовом процедуры LMViewAnalogAlarm(). При изменении статуса хоть одного из отслеживаемых параметров окно аналоговых тревог показывается автоматически. Настройка поведения этого окна производится в настройках проекта на вкладке «Аналоговые алармы»:



Окно аналоговых алармов выглядит следующим образом:



В заголовке окна указано общее количество аналоговых алармов. При активизации аларма в окно добавляется сформатированное сообщение следующего цвета:

«Желтый» - значение параметра в зоне «Предупреждение активно».

«Красный» - значение параметра в зоне «Тревога активна»

«Серый» - канал находится в состоянии какой-либо неисправности и его значение недостоверно.

### Методика создания аналогового аларма

В общем случае существует три способа для создания аналогового аларма:

1. Явное создание аларма в редакторе алармов. Для вызова редактора алармов выберите в меню «Ресурсы / Алармы...» и вкладку «Аналоговые алармы». Для каждого канала нужно добавить отдельный аларм со своими параметрами. Преимущество данного способа заключается в том, что созданные алармы работают всегда, независимо от того: отображается параметр на картах или нет.
2. Неявное создание аларма указанием в свойствах графического объекта карты одного из аналоговых алармов. Перед выполнением проекта для этого объекта карты неявно создается отдельный аналоговый аларм. Все настройки создаваемого аларма берутся из указанного аларма. Аналоговый аларм задается в окне свойств объекта карты на вкладке «Тревоги».
3. Объект «Прогресс бар» (полоса с заполнением) на вкладке «Тревоги» имеет параметры для изменения цвета заливки функционально схожие с параметрами аналогового аларма. Если установлена галочка «Добавить аналоговый аларм», то перед началом выполнения проекта будет неявно создан аналоговый аларм с указанными параметрами.

На следующем рисунке показано окно свойств объекта «Прогресс бар»:

Отображение	Программа	Бейсик	Тревоги	Звук	Связи
-------------	-----------	--------	---------	------	-------

☒ Изменять цвет заливки в зависимости от значения

Недопустимое повышение	<input type="text" value="90"/>
Повышение	<input type="text" value="80"/>
Понижение	<input type="text" value="10"/>
Недопустимое понижение	<input type="text" value="11"/>

☐ Добавить аналоговый аларм

Сообщение  
 ?

При сработке аларма проигрывается звуковой файл, заданный на вкладке "Звук"

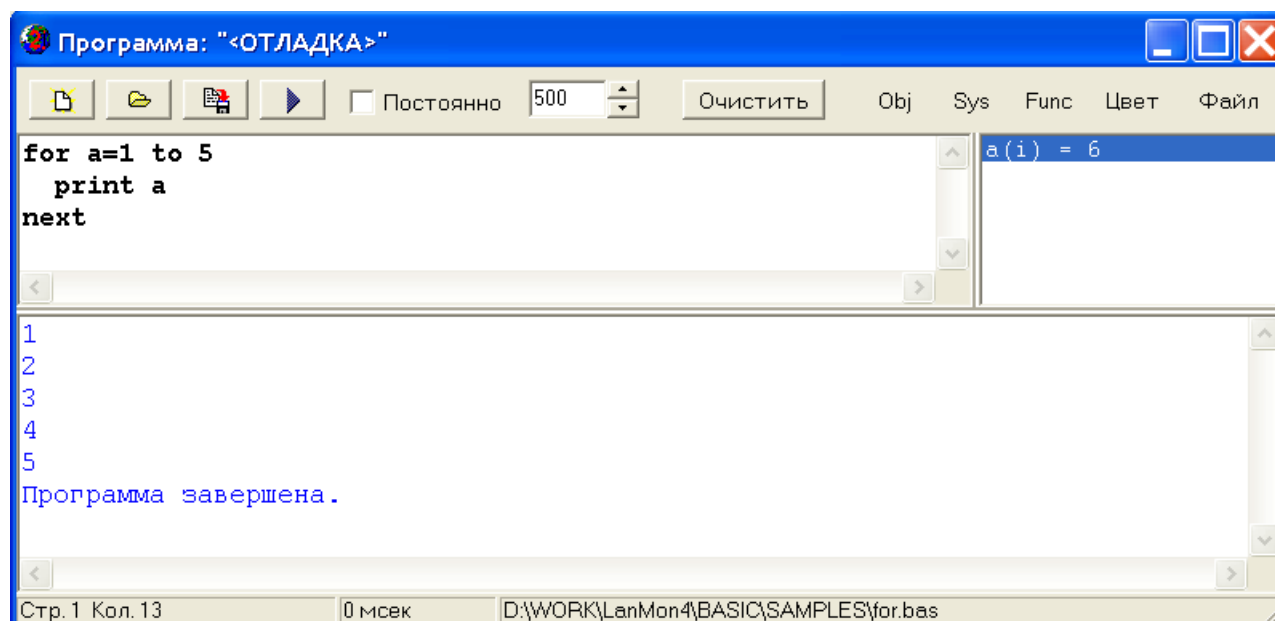
Описание аналоговых алармов хранятся в текстовом файле AALARM.INI в директории текущего проекта.

При изменении состояния аналогового аларма, вызывается событие в программе на скрипте: *OnAnalogAlarm*. Для добавления обработчика этого события в редакторе программ выберите в меню «События / Сработка аналогового аларма». Обработчик события получает параметры: номер сработавшего аларма *Alarm* и адрес канала, который вызвал сработку. Пример обработчика события приведем на C++Script:

```
// Это обработчик события при изменении состояния аналогового аларма
void OnAnalogAlarm(int Alarm, word A1, word A2, word A3, word A4)
{
    // Добавление строки с параметрами сработавшего аларма в lanmon.log
    LMProtokol( sprintf("Alarm=%u ADDR=%u.%u.%u.%u", Alarm, A1, A2, A3, A4) );
}
```

## Программы на бейсике

В APM LanMon 2, начиная с версии 0.81 появился встроенный язык программирования – интерпретатор бейсика. Он позволяет производить вычисления, манипулировать встроенными объектами APM LanMon и их свойствами. Для редактирования и отладки программ на бейсике используется специальный редактор программ. Для вызова редактора программ выберите пункт меню «Программа/Редактор программ на бейсике...», появится окно следующего вида:



В верхней части редактора программ расположена панель инструментов. Слева расположено окно ввода текста программ. Справа расположено окно со списком переменных бейсика. Снизу расположено окно вывода программ на бейсике (оператор «print» печатает именно туда). В строке статуса окна показано имя файла с текущей программой и время выполнения программы. Кнопки в панели инструментов выполняют следующие функции:

- Создать новую программу;
- Загрузить программу из файла;
- Сохранить программу в файл;
- Выполнить программу 1 раз;
- «Очистить» - Очистить окно вывода программы;
- «Obj» - Показать список объектов, доступных в данный момент;
- «Sys» - Показать список системных переменных;
- «Func» - Показать список функций бейсика;
- «Цвет» - Вставить цвет в текст программы (RGB закодированный в виде целого числа);
- «Файл» - Вставить имя файла в текст программы;

Список переменных используется для мониторинга значений переменных при отладке программы. Все переменные в APM LanMon глобальны. Переменные создаются при первом упоминании в программе и никогда не удаляются. Формат записи в списке переменных такой:

***имя\_переменной(тип переменной)=значение\_переменной***

Тип переменной может быть один из следующих: s – строковый, i – целый, f – с плавающей точкой. На картинке выше можно видеть переменную «a» целого типа со значением 6.

Галочка «Постоянно» позволяет совершать циклическое повторение выполнения программы с указанным интервалом времени.

Язык бейсик имеет ряд ограничений:

- Интерпретируемая структура языка: каждый раз при выполнении программы производится полный синтаксический анализ. Это требует значительных временных затрат и снижает общую производительность программы;
- Отсутствуют пользовательские процедуры и функции;
- Отсутствуют массивы;
- Обработчики событий не могут получать параметры;
- Каждый обработчик события находится в своем файле. В крупных проектах такой подход неоптимален;

Полное описание языка бейсик находится в документе: [basic.pdf](#). Надо отметить, что язык бейсик морально и функционально устарел и пользоваться им не рекомендуется. В АРМ LanMon 3 он поддерживается только для совместимости. Для программирования используйте скриптовые языки АРМ LanMon (см. следующую главу).

## Программы на скриптовых языках

В APM LanMon, начиная с версии 3.00, встроен новый скриптовый движок. Он значительно превосходит старый интерпретатор языка бейсик по скорости работы и функциональным возможностям. Скорость работы возросла от 5 до 15 раз, в зависимости от особенностей применения.

В версии APM LanMon 3 новый скриптовый движок существует параллельно со старым языком бейсик. Все старые проекты поддерживаются, т.е. совместимость снизу вверх полная. Тем не менее, старый бейсик надо считать морально устаревшим и в следующей версии APM LanMon (в версии 4) он не будет поддерживаться. Настоятельно рекомендуется все новые проекты делать с использованием скрипта и не пользоваться старым бейсиком.

Скриптовый язык имеет мультязычную архитектуру и позволяет программировать на четырех языках: C++, Pascal, Basic, Java script. Может быть выбран один из языков программирования. Специальная директива `#language` позволяет переключать язык по ходу выполнения программы. Это дает возможность, например, использовать функции, написанные на C++ в программе на Basic.

Скрипт имеет стандартный языковый набор:

- Переменные;
- Константы;
- Процедуры;
- Функции (с возможностью вложенности) с переменными/постоянными/умалчиваемыми параметрами;
- Стандартные операторы и объявления (включая `case`, `try/finally/except`, `with`);
- Стандартные типы данных: целый, дробный, логический, символьный, строковый, многомерные массивы, множество, `variant`;
- Классы (с методами, событиями, свойствами, индексами и свойствами по умолчанию);

APM LanMon имеет одну основную программу на скриптовом языке. Ее имя и частота выполнения задаются в настройках проекта. Директива `#include` языка C++ или `USES` языка Pascal позволяет включать в текст программы другие программы или модули. Основная программа содержит все процедуры, функции и обработчики событий проекта. Проект APM LanMon также может содержать выражения на скрипте (например, в динамических объектах карты).

Скрипт является компилируемым. Основная программа APM LanMon и все выражения на скрипте компилируется в псевдокод на этапе начала выполнения проекта, а затем псевдокод исполняется. Этим объясняется несколько возросшее время загрузки APM LanMon. В противовес этому, мы имеем скорость работы скрипта, значительно возросшую по сравнению со старым бейсиком.

Для написания программ на скрипте используется специальный редактор программ. Он встроен в APM LanMon. Редактор программ имеет функции пошагового исполнения программы с просмотром значений переменных после каждого шага выполнения. Для вызова редактора программ выберите пункт меню «Программа / Редактор программ...» или нажмите

кнопку .

В APM LanMon 3 встроен механизм отчетов. Каждый отчет тоже имеет свою, отдельную программу. Программа отчета «видит» все переменные и функции, описанные в основной программе.

Все выражения на скрипте также «видят» все переменные и функции, описанные в основной программе. Основное применение выражений на скрипте - в динамическом объекте карт APM LanMon.

Что осталось нереализованным в скрипте для любого языка:

- отсутствуют объявления типов (records, classes) в скрипте;
- нет записей (records), указателей (pointers), множеств (sets) (однако возможно использование оператора 'IN' - "a in ['a'..'c','d']");
- нет типа shortstrings;
- нет безусловного перехода (GOTO);
- нет указателей;

Реализация языка C++ имеет свою специфику:

- нет восьмеричных констант;
- нет 'break' в операторе SWITCH (SWITCH работает подобно Pascal CASE);
- операторы '++' и '--' возможны только после переменных, т.е. '++i' не будет работать; операторы '--', '++' и '=' ничего не возвращают, т.е. 'if(i++)' не будет работать;
- все идентификаторы не чувствительны к регистру;
- константа NULL это Null из Pascal- используйте nil вместо NULL;
- при инициализации массива значениями используются квадратные скобки вместо фигурных;
- нет указателей и оператора «->». Их роль выполняют ссылки на объекты и оператор «.» используется для доступа к свойствам и методам;

Подробное описание скриптовых языков программирования и редактора программ APM LanMon находится в документе «Руководство по программированию»: [script.pdf](#)

## Доступ к базам данных

В APM LanMon 3 появилась возможность по доступу к базам данных из программы на скрипте. Для этого в скрипте реализованы классы по работе с механизмом ADO:

*TADOConnection* = *class(TComponent)* – подключение к базе данных через ADO. В APM LanMon есть глобальное подключение к базе данных ADO. Параметры этого подключения задаются на вкладке «ADO» в настройках проекта. Есть глобальная переменная *SystemADOConnection: Class TADOConnection* которая обозначает это подключение.

*TADOQuery* = *class(TCustomADODataset)* SQL запрос к базе данных. Объект запроса можно динамически создавать оператор *new* (C++Script), например:

```
TADOQuery Query1;
```

```
Query1 = new TADOQuery(nil);
```

```
Delete Query1;
```


*TADOStoredProc* = *class(TCustomADODataset)* класс для выполнения хранимых процедур SQL сервера;

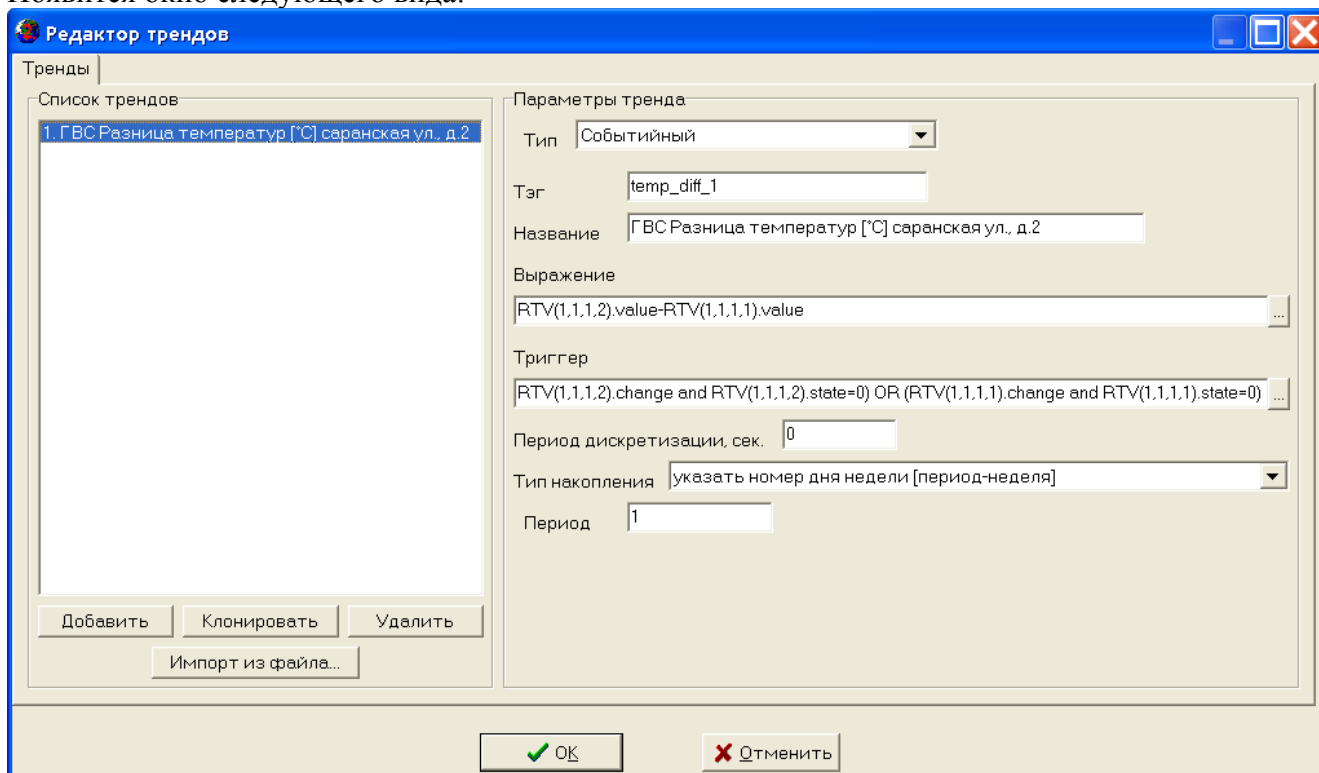
*TADOTable* = *class(TCustomADODataset)* класс для доступа к таблице базы данных;

Все перечисленные классы можно создавать динамически в основной программе APM LanMon или, статически в дизайнере отчетов. Объекты, определенные в дизайнере отчетов «видны» только для данного отчета.

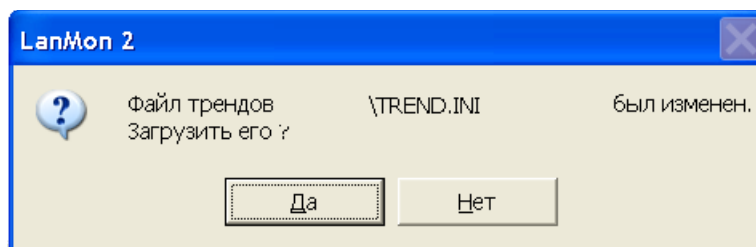
Пример доступа к Microsoft SQL серверу приводится в демонстрационном проекте, который идет в комплекте с APM LanMon.

## Тренды

Тренд в программе APM LanMon – это история изменения значения канала или выражения на встроенном бейсике. Тренд можно просматривать в виде графика и в виде списка значений в специальном окне. В одном окне может быть показано до 10 трендов (графиков). Добавление и изменение трендов производится в редакторе трендов. Для его вызова выберите пункт меню «Ресурсы/Тренды...» или нажмите кнопку  главной панели APM LanMon. Появится окно следующего вида:



В левой части окна расположен список трендов. При выборе одного из них, в правой части окна появляются его параметры. Все настройки трендов хранятся в файле `\trend.ini`. Этот файл имеет интуитивно-понятный текстовый формат и может редактироваться вручную текстовым редактором, например Notepad. После такого редактирования APM LanMon выдаст запрос:



На этот вопрос надо ответить «Да» и все изменения появятся в редакторе трендов.

Кнопка «Импорт из файла...» позволяет к существующему списку трендов добавить тренды из другого файла `trend.ini`

Тренды бывают двух типов *событийный* и *периодический*. В событийный тренд новое значение выражения помещается, если значение выражения триггера меняется из «ложь» в «истина» (или из 1 в 0). В событийный тренд значение выражения помещается с указанным *периодом дискретизации* (в секундах). *Тэг* тренда – это уникальное имя для доступа к тренду из



программы на бейсике. *Название* – название тренда, фигурирующее при отображении графика. *Выражение* – выражение на бейсике для значения тренда. *Триггер* – выражение на бейсике. В событийном тренде определяет момент помещения нового значения в тренд. Если задан триггер в периодическом тренде и значение его выражения «ложь» или 0, то помещение новых значений тренд приостанавливается до момента, когда значение выражения триггера станет «истина» или «не ноль». Период дискретизации – через сколько секунд помещать очередное значение в периодический тренд.

Тип накопления и период накопления определяют максимальный размер тренда перед помещением его в архив. Текущие данные тренда хранятся в файле `.\TREND\«тэг».001`. При превышении периода накопления текущие данные тренда помещаются в архив, т.е. в файл `.\TREND\«тэг».002`, а файл текущих данных `.\TREND\«тэг».001` начинается заново. Всего файлов тренда может быть 99: последний файл в архиве `.\TREND\«тэг».099`. Трактровка параметра «период накопления» зависит от параметра «тип накопления» следующим образом:

- глубина в отчетах («период накопления» – максимальное количество отчетов до помещения в архив)
- указать час [период - сутки] («период накопления» – час в который происходит помещение в архив данных тренда)
- указать номер дня недели [период-неделя] («период накопления» - 0...7 (0 или 7 это Воскресенье))
- указать день месяца [период-месяц] («период накопления» - номер дня в месяце 1..31)
- указать номера месяца [период-год] («период накопления» - номер месяца 1..12)

На картинке выше приведен пример событийного тренда по разнице температур воды в подающей и обратной трубах. Предположим 1.1.1.2 адрес канала подающей температуры, 1.1.1.1 адрес канала обратной температуры. Значение разницы температур добавляется в тренд если оно достоверно у обоих параметров (STATE=0) и если хоть один из параметров изменился. Тренд копится 1 неделю и помещается в архив. Период накопления указан – 1, это значит с первого дня недели – с понедельника. Функция бейсика «RTV» это более короткий синтаксис функции «RtVarByAddr». Свойство «change» принимает значение «истина» при изменении значения канала. После обращения к этому свойству оно автоматически принимает значение «ложь».

Для показа окна графиков и отображения тренда на график используются специальные функции бейсика. Вот, пример программы, которая присваивает графику номер 0 "trend1", а графику номер 1 "trend2". Затем показывает окно графиков.

```
TrendAssign("trend1",0)
TrendAssign("trend2",1)
TrendSetCaption("Графики по разнице подачи ГВС [T]")
TrendShowWindow(1)
```

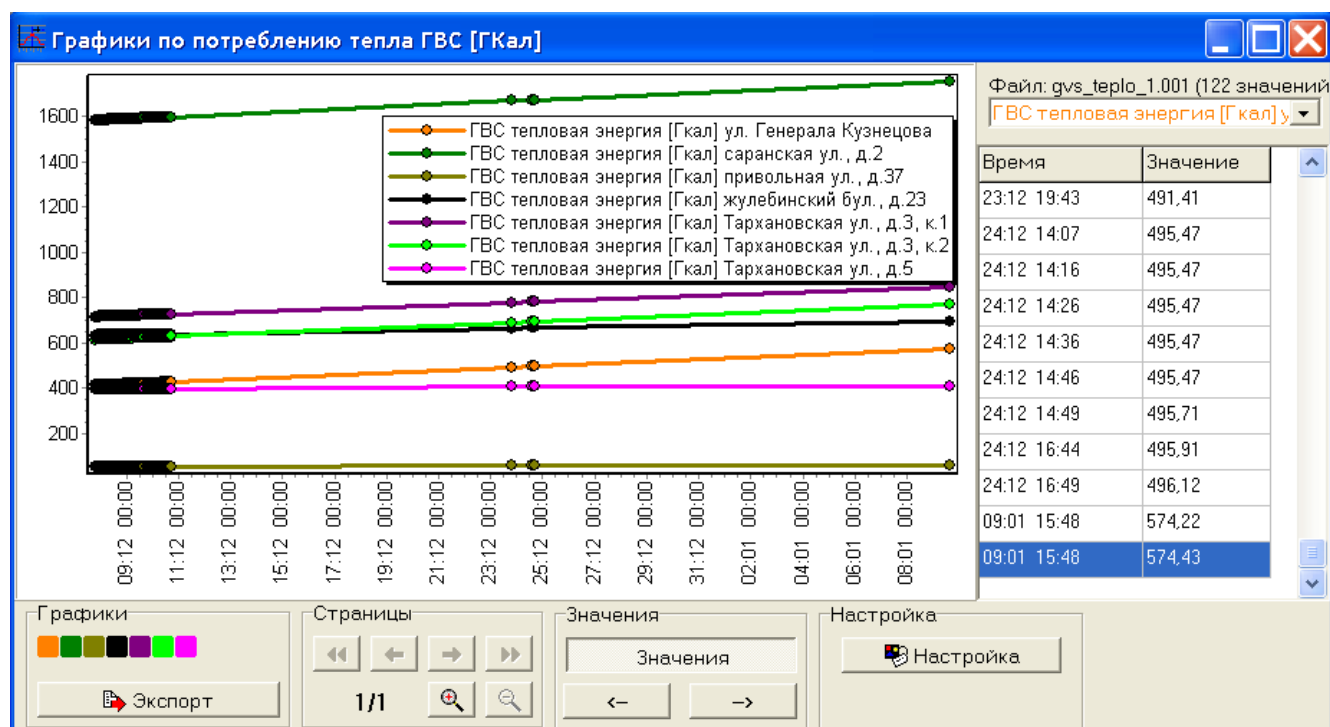
«trend1» и «trend2» это тэги трендов. Вместо тэга можно использовать индекс тренда в списке (с нуля). Номера графиков могут быть от 0 до 9. Т.е. всего можно отобразить одновременно 10 графиков. Функция *TrendAssign* отображает указанный тренд на указанном графике. Функция *TrendSetCaption* устанавливает заголовок окна просмотра графиков. Функция *TrendShowWindow(1)* показывает окно графиков на экране.

Вот пример программы для скрытия окна графиков и удаления ранее назначенных трендов.

```
TrendShowWindow(0)
TrendAssign(-1,0)
TrendAssign(-1,1)
```

*TrendAssign(-1,2)*  
*TrendAssign(-1,3)*  
*TrendAssign(-1,4)*  
*TrendAssign(-1,5)*  
*TrendAssign(-1,6)*  
*TrendAssign(-1,7)*  
*TrendAssign(-1,8)*  
*TrendAssign(-1,9)*

Окно графиков выглядит так:



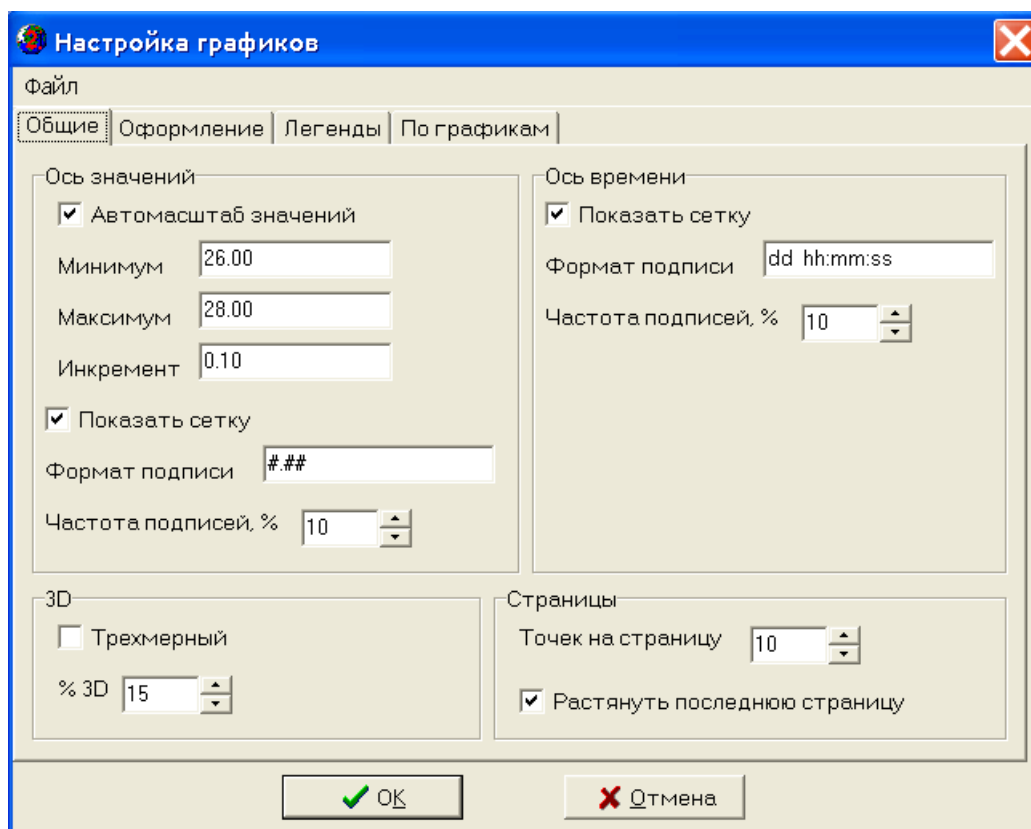
В верхней части окна расположены графики. Причем графики могут быть многостраничными. В правой части окна расположен список значений выбранного графика. В нижней части окна расположены органы управления просмотром графиков.

В разделе «Графики» каждому из 10 графиков соответствует кнопка того же цвета. Нажатие левой клавишей мыши на кнопке приводит к показу/сокрытию графика. Нажатие правой клавиши управляет показом метки со значением на графике. Кнопка «Экспорт» служит для экспорта текущего вида графиков в графический файл или в буфер обмена.

Раздел «Страницы» содержит кнопки для «листания» страниц графика, а также кнопки уменьшения и увеличения количества точек на одну страницу графика.

Раздел «Значения» служит для показа списка значений выбранного графика в правой части окна. Кнопка ← осуществляет переход на предыдущий период накопления из тренда (на архивные данные). Кнопка → вернуться к более позднему периоду накопления тренда.

В разделе «Настройка» одноименная кнопка вызывает окно настроек графиков. Окно настроек выглядит следующим образом:



Дадим пояснения лишь некоторым из настроек:

- Формат подписи по оси времени
  - «d» - день 1-31
  - «dd» - день 01-31
  - «ddd» - краткое трехбуквенное название дня недели
  - «dddd» - полное название дня недели
  - «m» - месяц 1-12
  - «mm» - месяц 01-12
  - «mmm» - краткое трехбуквенное название месяца
  - «mmmm» - полное название месяца
  - «yy» - две цифры года 00-99
  - «yyyy» - четыре цифры года 0000-9999
  - «h» - часы 0-23
  - «hh» - часы 00-23
  - «n» - минуты 0-59
  - «nn» - минуты 00-59
  - «s» - секунды 0-59
  - «ss» - секунды 00-59
- Формат подписи по оси значений может содержать следующие символы:
  - «0» - место цифры должно быть дополнено нулем
  - «#» - место цифры
  - «.» - место десятичной точки
  - «,» - если в маске есть хоть 1 запятая, то слева от десятичной точки тысячи будут разделяться запятой.
  - «;» - разделяет маски. Можно указать свою маску для положительных, отрицательных и нулевых чисел.
  - «E+» - использовать экспоненциальный формат чисел

Символы заключены в одинарные или двойные кавычки остаются без изменений

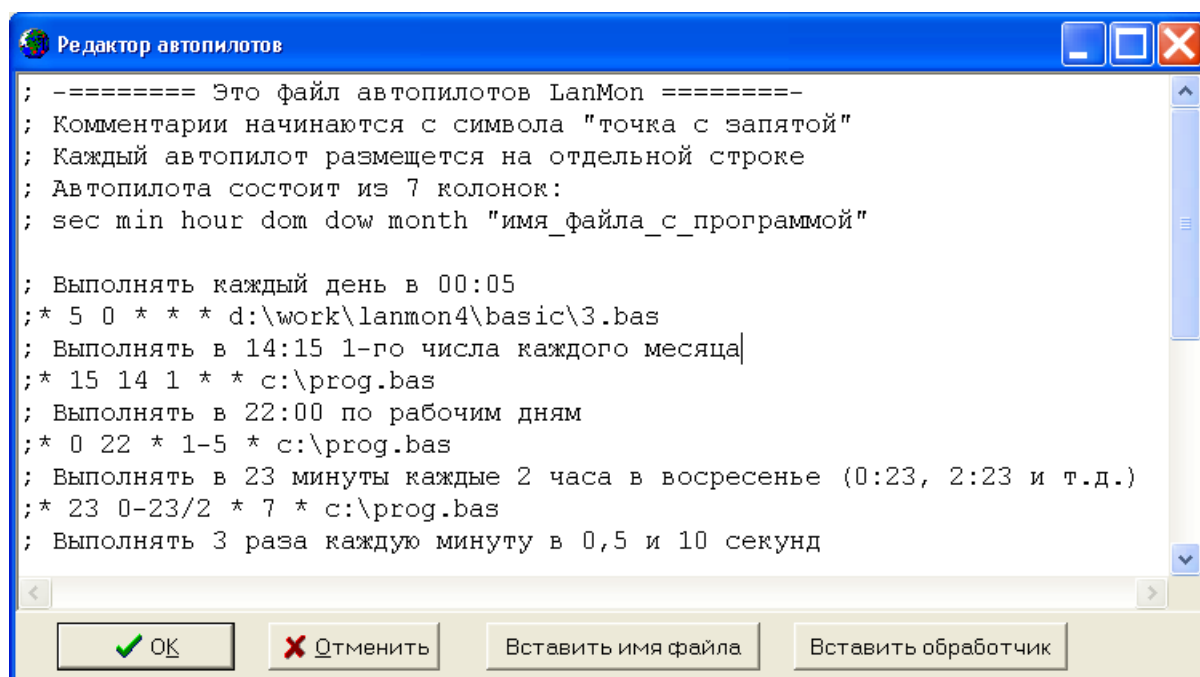
#### Пример

	Маска		Число		
	1234	-1234	0.5	0	
0	1234	-1234	1	0	
0.00	1234.00	-1234.00	0.50	0.00	
###	1234	-1234	.5		
###0.00	1,234.00	-1,234.00	0.50	0.00	
###0.00;(###0.00)	1,234.00	(1,234.00)	0.50	0.00	
###0.00;;Zero	1,234.00	-1,234.00	0.50	Zero	
0.000E+00	1.234E+03	-1.234E+03	5.000E-01	0.000E+00	
####E-0	1.234E3	-1.234E3	5,00E-01	0,00E+00	

Из меню «Файл» можно сохранить настройки графиков в файл с расширением .grf. Рекомендуется сохранять настройки графиков в директорию ./TEMPLATE/. Функция бейсика TrendLoadParam(“имя файла.grf”) позволяет прочесть настройки графиков. Если имя файла указано без пути, то производится поиск в поддиректории ./TEMPLATE/. Описанный механизм удобно применять для отображения разных физических величин. Например, для отображения температур свои настройки, а для отображения давления свои.

## Автопилоты

Автопилот – это периодически выполняемое действие. Для настройки автопилота надо определить время выполнения программы. Автопилоты настраиваются путем написания текстового файла в специальном редакторе. Для добавления или изменения автопилота выберите пункт меню «Ресурсы/Автопилоты...» - появится окно следующего вида:



В редакторе автопилотов приведены примеры автопилотов. Каждый автопилот описывается строкой, состоящей из семи колонок. Строка автопилота должна завершаться возвратом каретки. Колонки должны разделяться пробелами или табуляцией. В следующей таблице приведено описание колонок автопилота:

№	Колонка	Возможные значения
1	Секунды	0...59
2	Минуты	0...59
3	Часы	0...23
4	День месяца	1...31
5	День недели	0...7 (0 или 7 это Воскресенье)
6	Месяц	1...12
7	Имя файла с программой	Имя файла с программой на бейсике LanMon (с расширением .bas) или имя функции-обработчика на скрипте. Когда подходит время автопилота эта программа выполняется. Если указано имя программы без пути, то поиск программы осуществляется в стандартной поддиректории для программ на бейсике проекта APM LanMon: ..\BASIC\

Любая колонка, кроме имени файла программы, может быть символ (\*). Это означает – любое значение.

Допускаются диапазоны чисел, например «8-11» это означает 8, 9, 10 и 11.

Допускаются списки. Списки это диапазоны, разделенные запятой. Например: «1,2,5,9» или

«0-4,8-12».

Допускается указывать шаг значений после диапазона. Например: «0-23/2» эквивалентно «0,2,4,6,8,10,12,14,16,18,20,22». Шаг можно указывать после звездочки. Например, чтобы сказать «каждые 2 часа» укажите: «\*/2».

Примечание: день выполнения программы может быть одновременно задан днем недели и днем месяца. Например: «\* 30 4 1,15 5 \* c:\prog.bas» - программа prog.bas будет выполняться в 4:30 утра 1-го и 15-го числа каждого месяца и в каждую пятницу.

Автопилот может вызвать программу на LanMon бейсике из отдельного файла с расширением bas или функцию из основной программы на скрипта с указанным именем и одним параметром типа Integer. Параметр означает номер автопилота в списке (с нуля), вызвавший функцию. Рекомендуется использовать второй вариант (функцию скрипта) как значительно более быстрый и функциональный. LanMon бейсик на данный момент морально и функционально устарел.

Для вставки имени файла программы на бейсике в строку автопилота удобно пользоваться кнопкой «Вставить имя файла». Если имя файла программы уже задано в строке автопилота, данная кнопка позволяет просмотреть и отредактировать программу в редакторе программ.

Для вставки имени функции скрипта нажмите кнопку «Вставить обработчик». Тип программы различается по наличию расширения «.bas».

Для сохранения списка автопилотов нажмите клавишу «ОК». Для отмены произведенных изменений нажмите клавишу «Отменить».

## Отчеты RTF

АРМ LanMon позволяет формировать отчеты по значениям каналов или любым выражениям встроенного языка бейсик. Для создания отчета надо подготовить его шаблон. Шаблон отчета это файл с форматированным текстом в формате RTF. В текст шаблона в квадратных [] или фигурных {} скобках вставляются выражения на LanMon бейсике. При создании отчета из шаблона выражения в фигурных скобках вычисляются и заменяются на их значения. Выражения на бейсике могут использовать любые функции и операции LanMon бейсика. Шаблон отчета может содержать таблицы. Не все редакторы текста в формате RTF поддерживают таблицы. Для создания шаблона рекомендуется использовать Microsoft Word. Шаблоны отчетов должны храниться в стандартной поддиректории проекта для шаблонов: ..\Report\

АРМ LanMon имеет встроенный редактор шаблонов отчетов. Этот редактор позволяет создавать текст шаблона отчета и просматривать готовые отчеты. Есть ограничения: таблицы не поддерживаются. Для вызова редактора выберите пункт меню «Отчеты / Отчеты RTF...».

Пример шаблона отчета приведен ниже:

### Сводная таблица по теплу 12 м-на за {sprintf("%02u.%02u.%u", @day, @month, @year/1)}

	Отопление				Горячее водоснабжение			
	Температура, °C		Давление, Бар		Температура, °C		Давление, Бар	
	Подающая	Обратная	Подающая	Обратная	Подающая	Обратная	Подающая	Обратная
1201	[RtVarByAddr(1,5,1,712).value]	[RtVarByAddr(1,5,1,713).value]	[RtVarByAddr(1,5,1,714).value]	[RtVarByAddr(1,5,1,715).value]	[RtVarByAddr(1,5,1,702).value]	[RtVarByAddr(1,5,1,703).value]	[RtVarByAddr(1,5,1,704).value]	[RtVarByAddr(1,5,1,705).value]
1202	[RtVarByAddr(1,5,2,712).value]	[RtVarByAddr(1,5,2,713).value]	[RtVarByAddr(1,5,2,714).value]	[RtVarByAddr(1,5,2,715).value]	[RtVarByAddr(1,5,2,702).value]	[RtVarByAddr(1,5,2,703).value]	[RtVarByAddr(1,5,2,704).value]	[RtVarByAddr(1,5,2,705).value]
...								

Результат, полученный при обработке этого шаблона в редакторе отчетов:

	Отопление				Горячее водоснабжение			
	Температура, °C		Давление, Бар		Температура, °C		Давление, Бар	
	Подающая	Обратная	Подающая	Обратная	Подающая	Обратная	Подающая	Обратная
1201	23.89	22.57	42949648.00	3.66	0.00	0.00	0.00	0.00
1202	75.27	70.67	7.78	6.85	26.77	26.72	42949648.00	0.40
...								

В заголовке отчета производится форматирование текущей даты функцией бейсика *sprintf(...)*. В колонки таблицы отчета выводятся текущие значения каналов LanMon при

помощи функции *RtVarByAddr(...).value*. Функция *RtVarByAddr(...)* возвращает ссылку на объект типа «Канал LanMon» по его адресу и мы используем его свойство *value*, которое дает значение канала. Функция *RTV(...)* представляет собой сокращенный вариант функции *RtVarByAddr(...)*. При редактировании шаблонов отчетов некоторыми редакторами (например, Microsoft Word) формат файла RTF может оказаться непригодным для создания отчета. Для решения этой проблемы служит процедура нормализации шаблонов отчетов. Она вызывается через пункт меню «Отчеты/Нормализовать отчеты RTF». Эта процедура нормализует все шаблоны отчетов, находящиеся в стандартной директории для шаблонов. Выполняйте процедуру нормализации после редактирования шаблона отчета средствами, отличными от стандартного редактора APM LanMon.

Для работы с отчетами в LanMon бейсике предназначены следующие функции:

- *rptRun("файл\_шаблона","файл\_отчета")* формирует отчет из шаблона.
- *rptPrint("файл\_отчета ")* печатает отчет на принтере по умолчанию.
- *rptShow("Заголовок окна","файл\_отчета")* показывает отчет на экране в новом окне с указанным заголовком.

Если у файла шаблона не указан путь, он будет искаться в стандартной поддиректории для шаблонов отчетов. Пример программы по работе с отчетами можно найти в любом проекте APM LanMon в файле `..\BASIC\SAMPLES\report.bas`



## Генератор отчетов (новый)

В APM LanMon 3 встроен новый генератор отчетов значительно более функциональный, чем отчеты RTF. Для вызова редактора отчетов выберите в меню «Отчеты/Редактор отчетов...». Генератор имеет следующие возможности:

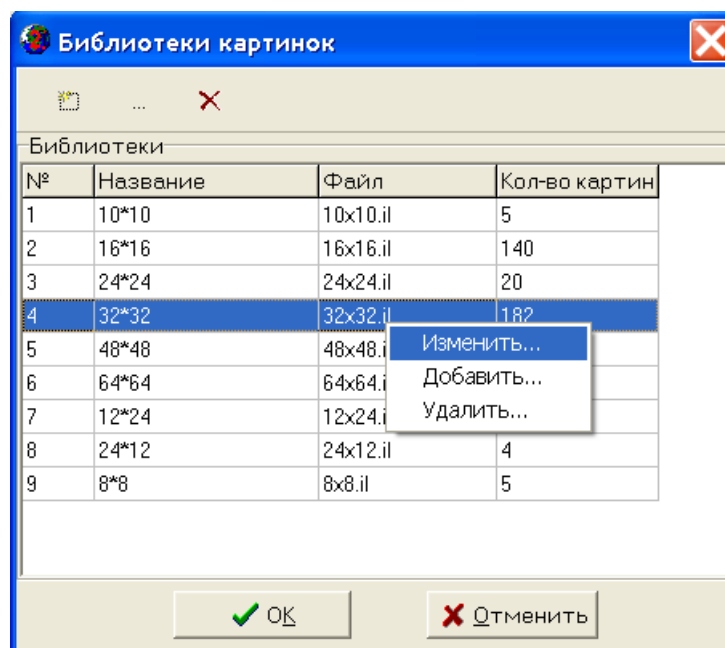
- Для построения отчета может использоваться любой источник данных (SQL сервер) через механизм ADO;
- В качестве исходных данных отчета возможно применение значений переменных, массивов, функция и свойств объектов APM LanMon;
- В каждый отчет встроена отдельная программа на скрипте. Поддерживаются 4 языка - PascalScript, C++Script, JScript, BasicScript. Программа в отчете используется для управления процессом построения отчета;
- Отчет может содержать диалоговые формы для запроса параметров перед построением. В отчете может быть несколько диалоговых форм;
- Редактор скрипта с подсветкой синтаксиса;
- Экспорт готового отчета в TXT, RTF, HTML, PDF, XLS, XML, JPG, BMP, TIFF форматы;
- Бэнд-ориентированный генератор отчетов;
- Встроенный мощный дизайнер, доступный и в run-time: полный WYSIWYG, preview как в MS Word;
- Многостраничные отчеты; составные (композиционные) отчеты; вложенные отчеты;
- Группы; многоколоночные отчеты; master-detail-detail отчеты;
- cross-tab отчеты; двухпроходные отчеты; "живые" отчеты;
- Набор наиболее популярных компонентов: Текст, Линия (в том числе и диагональная), Рисунок, Фигура, OLE объект, RichText, RX Rich 2.0,
- Вращение текста под углом 0..360;
- Полное Undo/Redo в дизайнера отчетов;
- Редактирование сформированного отчета;

Подробное описание отчетов содержится в документе «Руководство по генератору отчетов» report.pdf.

Примеры отчетов включены в состав демонстрационных проектов, которые идут в комплекте с APM LanMon 3.

## Библиотеки картинок

Библиотека картинок это перечень графических картинок одинаковой размерности. Библиотека имеет следующие параметры: название, ширину и высоту хранимых картинок в экранных точках (пикселях). На диске библиотека хранится в двоичном файле специального формата с расширением «.IL». По умолчанию, все файлы библиотек должны храниться в поддиректории проекта: «..\IMAGES\». Если у файла библиотеки не указан полный путь, то подразумевается, что она находится в стандартной директории. Библиотека картинок хранит картинки для отображения в объекте карты «Набор картинок». Библиотек картинок может быть несколько: для хранения картинок разных размеров. Для создания и редактирования библиотек картинок выберите пункт меню «Ресурсы/Библиотеки картинок...» - появится окно следующего вида:



Список библиотек имеет колонки, назначение которых очевидно. В верхней части окна расположена панель с кнопками:

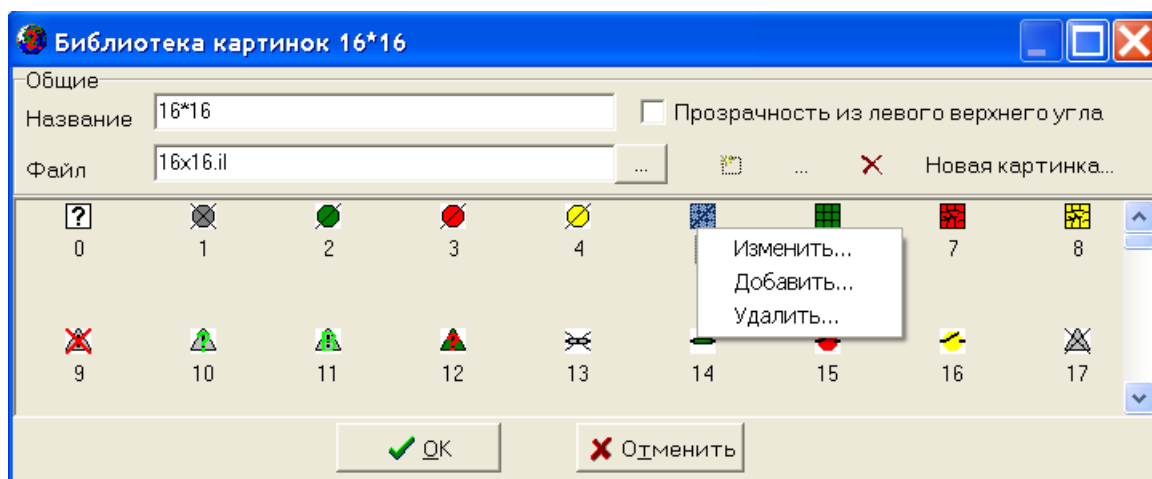
- создание новой библиотеки картинок и добавление ее в список библиотек. Перед созданием библиотеки будет выдан запрос о вводе размеров картинок, которые будет хранить библиотека. Размер картинок измеряется в экранных точках (пикселях).

- редактирование имеющейся библиотеки (добавление, изменение, удаление картинок).

- удаление библиотеки из списка и ее файла с диска.

Для сохранения произведенных изменений в списке библиотек нажмите кнопку «ОК». Для отмены – кнопку «Отменить».

При выборе кнопок или появится окно редактора картинок выбранной библиотеки. Оно выглядит так:



В заголовке окна указаны реальные размеры картинок в библиотеке. Каждая картинка имеет номер (первая картинка номер 0 и т.д.). В верхней части окна кроме названия библиотеки и файла, в котором хранятся ее картинки, расположены следующие кнопки:

- добавление картинки из файла BMP или ICO в конец библиотеки;

- заменить текущую картинку на картинку из файла BMP или ICO;

- удаление последней картинки в библиотеке (для удаления картинка должна быть выбрана).

Данное ограничение введено намеренно: отображаемые объекты карты ссылаются на картинки в библиотеке по ее номеру (индексу). Например: в библиотеке 10 картинок. Мы удалим вторую. Номера картинок после второй сдвинутся вниз (т.е. 3-я станет 2-ой и т.д.). Это может привести к тому, что придется заново настраивать отображаемые объекты. Поэтому ненужную картинку лучше заменить на нужную, а не удалять.

**«Новая картинка...»** - данная кнопка загружает стандартный графический редактор Windows – Paint. Вы можете нарисовать картинку и сохранить ее на диске для добавления в библиотеку.


**«Прозрачность из левого верхнего угла»** - если эта галочка установлена: прозрачный цвет картинки берется из левого верхнего пикселя картинки. Если галочка не установлена, прозрачным считается цвет (Olive).

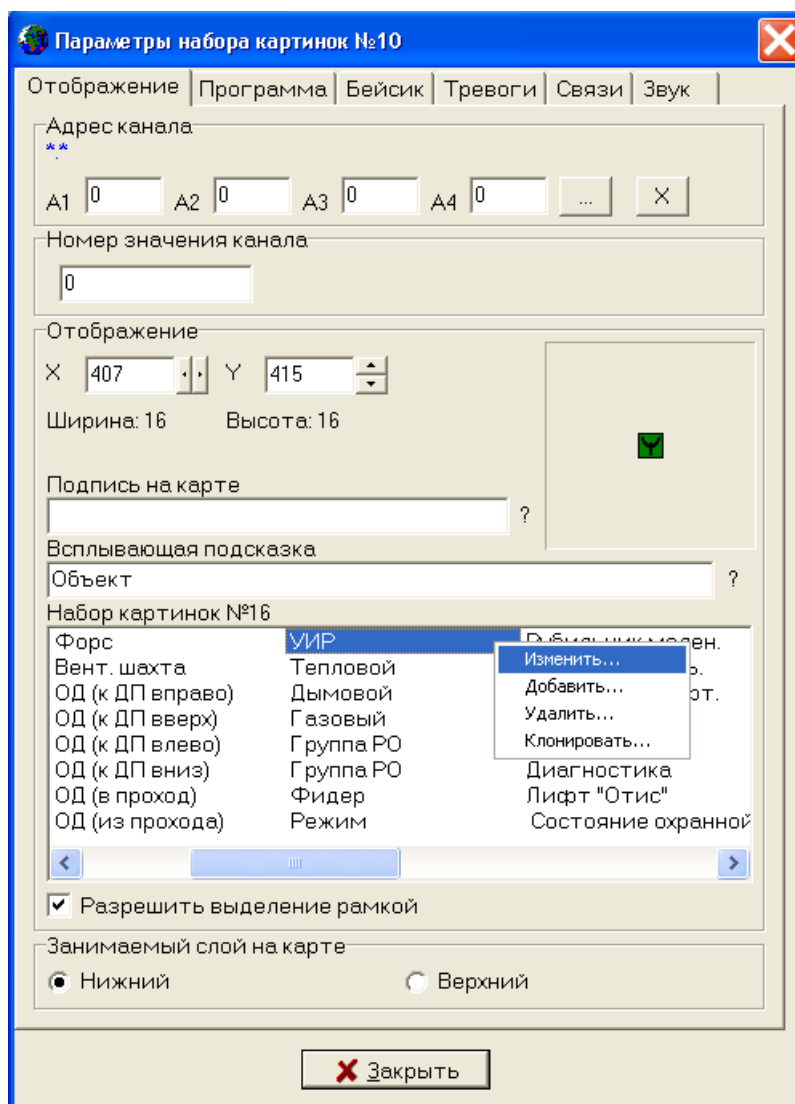
Для сохранения всех изменений в картинках библиотеки нажмите кнопку «ОК». Для отмены – кнопку «Отменить».

## Наборы картинок

Набор картинок это отображаемый объект карты. Каждому значению канала или диапазону значений соответствует определенная картинка из библиотеки. APM LanMon включает стандартные библиотеки картинок для отображения состояния типовых извещателей и объектов. Можно создавать пользовательские наборы картинок.

В системах с большим числом дискретных параметров набор картинок является основным объектом карт. Например, в системах охранной, пожарной и газовой сигнализации, каждое состояние извещателя отображается на карте определенным значком.

Кнопка  панели объектов карт APM LanMon позволяет добавить набор картинок на карту. Для настройки параметров набора картинок дважды щелкните левой кнопкой мыши на наборе картинок на карте. Появится окно следующего вида:



Самый первый набор в списке называется «Числовые типы». Он используется для отображения качества каналов LanMon с числовым типом и не должен изменяться.

В списке «Набор картинок» можно выбрать один из стандартных наборов. Для изменения стандартного набора или создания нового, нажмите правую клавишу мыши для вызова контекстного меню и выберите нужно действие.

Для добавления нового набора картинок выберите наиболее похожий по смыслу из списка и в контекстном меню выберите «Клонировать...». При этом будет создан новый набор картинок на основе выбранного.

При добавлении или изменении списка картинок появится окно следующего вида:

**Редактор набора картинок**

Общие  
 Название:  Тип канала LanMon по умолчанию:

Состояния

Качество канала	Операция	Значение	Библиотека	Картинка	Кол-во	Название состояния
OK	?	0	16*16		1	Неизвестное состояние
Выключен	?	0	16*16		1	Выключен
Состояние не определено	?	0	16*16		1	Состояние не определено
Неисправно устройство	?	0	16*16		1	Неисправен датчик
Неисправен контроллер	?	0	16*16		1	Неисправен контроллер
Значение недостоверно	?	0	16*16		1	Значение недостоверно
Не подключен	?	0	16*16		1	Датчик не подключен
OK	=	0	16*16		1	Норма
OK	=	1	16*16		2	Сработка

Параметры состояния

Название:  Библиотека картинок:  ...

Качество:  Кол-во картинок:

Операция:  Картинка:

Значение:

Набор картинок состоит из состояний. При изменении значения канала, APM LanMon ищет подходящее состояние в списке состояний, начиная со второго. Первое состояние в списке – это состояние по умолчанию. Сначала сравнивается качество канала. Если качество совпало – поиск завершается.

Если качество «OK» - выполняем следующее действие: «текущее значение канала» «Операция» «Значение». Если результат такого действия «истина» или «не ноль» - поиск прекращается. Текущее значение канала приводится к целому типу. «Значение» - число целого типа.














Если ни одно из состояний не подошло – принимаем состояние по умолчанию (первое в списке).

Кнопка добавляет новое состояние в конец списка состояний. Кнопка удаляет текущее состояние. Для каждого состояния задается также библиотека картинок, картинка в библиотеке и количество картинок. Для редактирования выбранной библиотеки нажмите кнопку . Если у состояния задано больше одной картинки – они будут анимироваться (сменяться по кругу каждые 200 мсек).

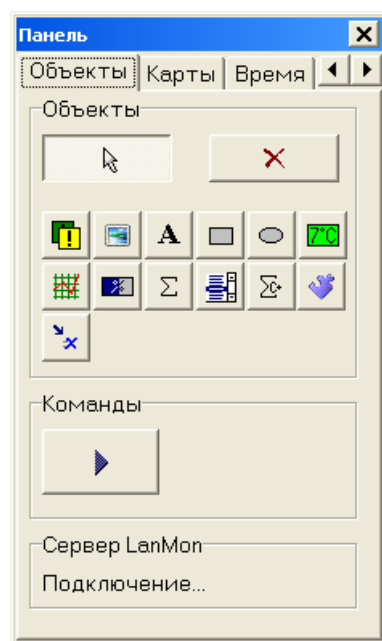
Для сохранения всех изменений в наборе картинок нажмите кнопку «OK». Для отмены – кнопку «Отменить».

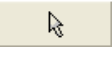

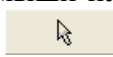
## Карты


Карта это форма, на которой размещаются отображаемые объекты. Карта, стоящая первой в списке автозагрузки в настройках проекта является главной, т.е. после загрузки проекта в АРМ LanMon она автоматически появляется на экране. Карта может содержать следующие отображаемые объекты:

-  Набор картинок – каждому значению канала или диапазону значений соответствует определенная картинка из библиотеки. АРМ LanMon включает стандартные библиотеки картинок для отображения состояния типовых извещателей и объектов. Можно создавать пользовательские наборы картинок (смотри главу «Наборы картинок»).
-  Картинка – на карте можно разместить картинку в формате BMP.
-  Текст – текстовая метка на карте. Может иметь свой цвет, шрифт и рамку. Может отображать значение канала или прорисовываться из программы.
-  Прямоугольник – имеет границу и заливку. Параметры отображения и координаты на экране могут изменяться из программы.
-  Эллипс – то же что и прямоугольник.
-  Температура воздуха – текстовая метка для отображения значения температуры воздуха. Цвет заливки меняется в зависимости от значения температуры.
-  График – отображение значения канала во времени. Настраиваются цвета, шаг сетки и скорость обновления.
-  Прогресс-бар – может отображать значение канала в виде прогресс-бара в установленном диапазоне значений.
-  Динамический объект на бейсике. Имеет возможность отображения текста, картинки и прогресс-бара.
-  Список строк. Добавлять строки можно на этапе редактирования карты и динамически из скрипта.
-  Динамический объект на одном из скриптовых языков. Язык объектов данного типа задается в параметрах карты. Имеет возможность отображения текста, картинки и прогресс-бара.
-  Замкнутая область произвольного вида. Состоит из прямых отрезков. Можно управлять цветом заливки и цветом границы области из программы на скрипте.
-  Объект ActiveX. Вы можете разместить на карте любой из объектов, зарегистрированных в системе.

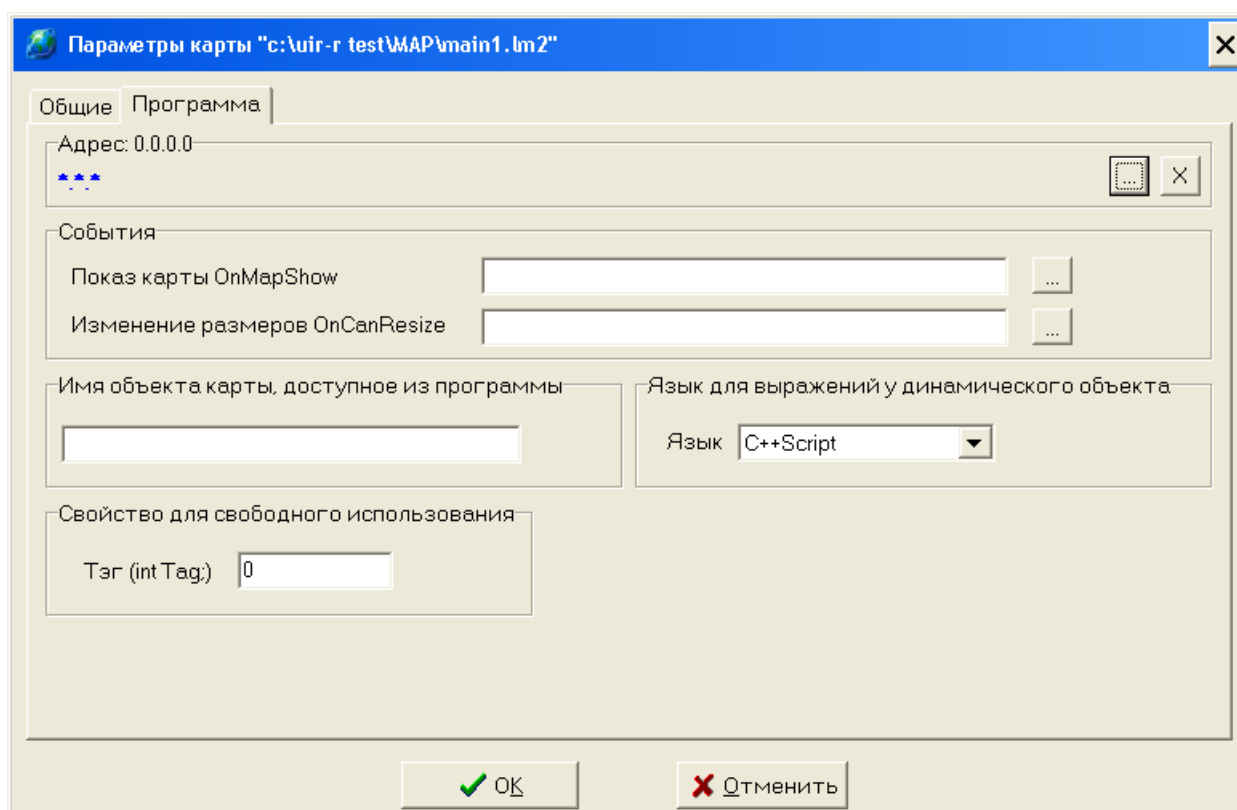
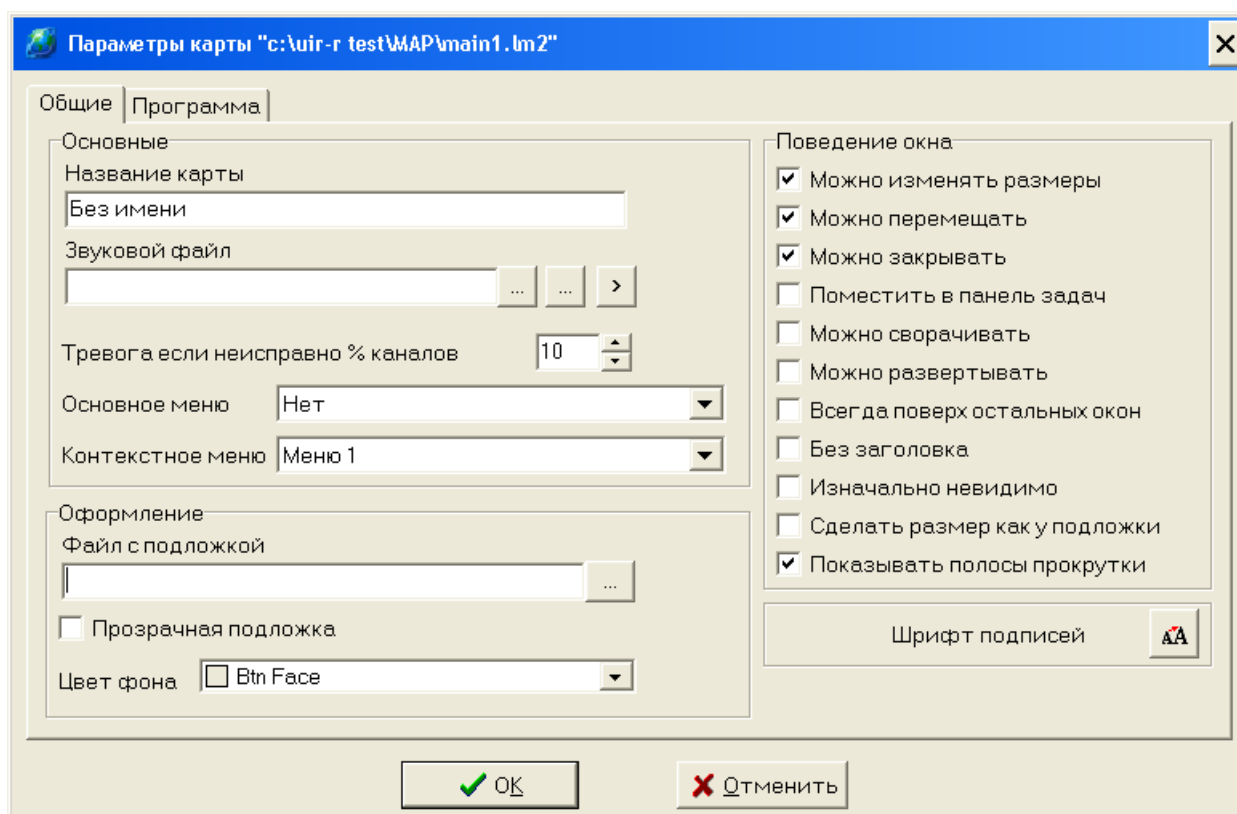
Добавление на карту отображаемого объекта производится при помощи панели объектов. Если панель отсутствует на экране выберите пункт главного меню «Просмотр / Показать панель». Панель выглядит следующим образом:



Для добавления объекта на карту выберите объект кнопкой в секции «Объекты» и щелкните левой клавишей мыши на пустом месте карты. После добавления объекта нажмите кнопку . Альтернативный способ добавления объекта на карту – перетаскивание канала из дерева каналов. При перетаскивании ветви каналов на карту – добавляются все каналы из ветви. Для удаления объекта нажмите клавишу  и щелкните левой клавишей мыши на объектах карты, которые надо удалить. После удаления всех объектов нажмите кнопку .

Кнопка  служит для запуска проекта на выполнение. Она используется при отладке проекта. Также в панели отображается информация о состоянии подключения к серверу LanMon.

Карта имеет верхнее меню и контекстное меню, открывающееся щелчком правой клавишей мыши. Эти меню предназначены для редактирования карты и недоступны в режиме выполнения проекта. Редактирование настроек карты доступно через окно параметров. Для открытия окна параметров карты выберите из меню пункт «Параметры карты...» - откроется окно следующего вида:



«**Название карты**» - название, отображаемое в строке заголовка карты (свойство String TMap::Name).

«**Звуковой файл**» - привязка звукового WAV файла к данной карте. Имя этого файла доступно из программы через свойство карты String TMap::Sound.



«Тревога если неисправен % каналов» - если указанный процент каналов на карте имеет статус неисправности, то активируется аларм, заданный в настройках проекта на вкладке «Дискретные алармы». Если задано 0% - активация аларма не выполняется никогда.

«Основное меню» - Выбор меню окна карты. Меню должно быть предварительно настроено. Настройка меню выполняется из меню главного окна «Ресурсы \ Меню карты».

«Контекстное меню» - Выбор всплывающего меню карты (по нажатию правой клавиши мыши). Меню должно быть предварительно настроено. Настройка меню выполняется из меню главного окна «Ресурсы \ Контекстное меню».

«Файл с подложкой» - в качестве подложки карты можно назначить файл с картинкой типа BMP.

«Прозрачная подложка» - сделать фон подложки прозрачным. Цвет прозрачности берется у левого нижнего пикселя подложки.

«Цвет фона» - указать цвет подложки.

«Шрифт подписей» - по умолчанию отображаемый объект «набор картинок» имеет подпись на экране этим шрифтом.

В разделе «Поведение окна» задается особенности поведения окна карты.

«Адрес» - карте должен быть назначен канал LanMon с типом «карта». При изменении состояния охраны карты в журнал событий будут писаться события от этого канала.

«Событие OnMapShow» - можно добавить в программу на скрипте обработчик события показа карты на экране.

«Событие OnCanResize» - можно добавить в программу на скрипте обработчик события на изменение размеров карты.

«Имя объекта карты, доступное из программы» - В программе будет доступна глобальная переменная – объект типа TMap. Можно вызывать его методы и свойства для манипуляции с картой.

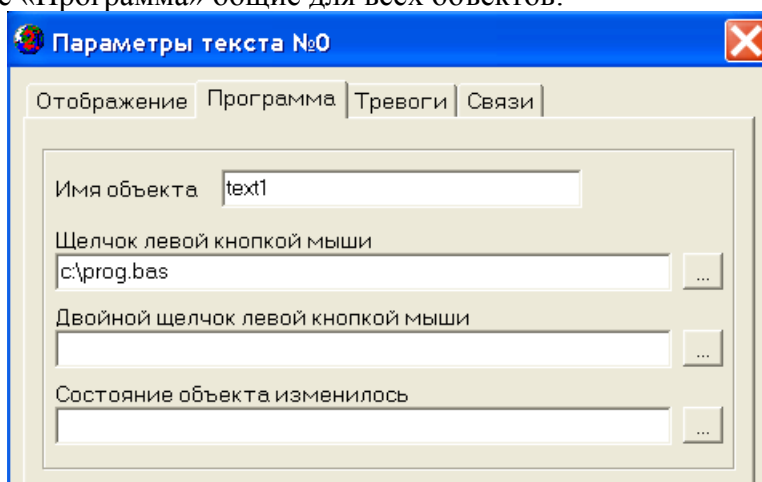
«Язык выражений карты» - задайте язык программирования, на котором будете писать выражения у динамического объекта на скрипте.

«Тэг» - свойство карты int Tag. Для произвольного использования программистом. Начальное значение этого свойства можете задать в этом поле.

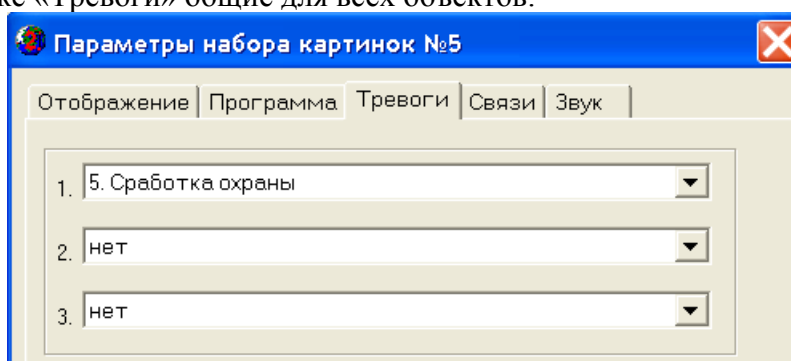
Для подтверждения изменений в параметрах карты нажмите кнопку «ОК». Параметры карты хранятся в файле карты. Для сохранения параметров карты на диске сохраните карту.

## Отображаемые объекты карты

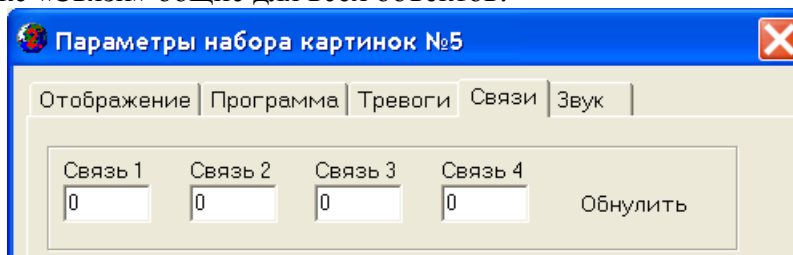
Каждый отображаемые объекты на карте имеет редактор свойств. Для его вызова дважды щелкните левой клавишей мыши на объекте. Есть свойства общие для всех объектов. Свойства на вкладке «Программа» общие для всех объектов:



Задается имя объекта, доступное из программы на бейсике. Имя файла с программой, вызываемой по щелчку левой клавишей мыши и по двойному щелчку левой клавишей мыши. Свойства на вкладке «Тревоги» общие для всех объектов:



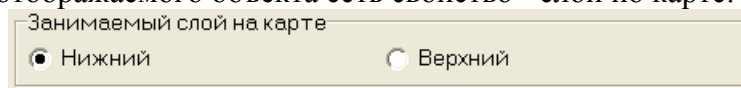
Для каждого отображаемого объекта можно задать 3 аларма. Эти алармы будут обрабатывать значения канала, привязанного к отображаемому объекту. Алармы настраиваются в редакторе алармов. Если алармы не нужны – надо задать «нет» для всех алармов. Свойства на вкладке «Связи» общие для всех объектов:



Для каждого отображаемого объекта можно назначить от одной до четырех связей. Механизм назначения связей и алгоритм их работы описан в разделе «Ресурсы/Связи». Свойства на вкладке «Отображение» общие для всех объектов:

- Адрес канала LanMon
- Координаты центра отображаемого объекта на экране X, Y
- Разрешить выделение рамкой – при наведении курсора мыши на объект в режиме выполнения проекта он обводится рамкой
- Занимаемый слой на карте – объект может находиться в одном из двух слоев на карте: нижнем или верхнем

Отображаемые объекты могут пересекаться. На карте есть 2 слоя объектов: верхний и нижний. У каждого отображаемого объекта есть свойство - слой на карте:



Для отображаемых объектов карты предусмотрены групповые операции. Выделение объектов производится движением мыши с нажатой левой клавишей. При этом все объекты, попавшие в получившийся прямоугольник, будут выделены. Альтернативный способ выделения: удерживая клавишу «Ctrl» щелкните левой клавишей мыши на объекте карте. Выделенные объекты обведены красной рамочкой. Для снятия выделения объектов щелкните левой клавишей мыши на пустом месте карты.

После выделения объектов станут доступны такие операции как: копирование, связывание и выравнивание. Поддерживаются групповое изменение свойств объектов карты: выделите однотипные объекты и выведите окно свойств любого из них. При изменении одного из свойств оно одинаково изменится во всех выделенных объектах.

### Свойства уникальные для разных объектов



Набор картинок имеет следующие свойства:

- «Подпись на карте» – текстовая строка, привязанная к отображаемому объекту;
- «Отображение» – в списке существующих наборов картинок выберите нужный;



Картинка имеет следующие свойства:

- «Всплывающая подсказка» – текстовая строка. Возможны специальные подстановки аналогичные всплывающей подсказке в параметрах карты;
- «Картинка» – путь к файлу картинки (BMP). По умолчанию файл картинки берется из стандартной поддиректории проекта для картинок .. \BMP\;



Текст имеет следующие свойства:

- «Текст» – отображаемый на экране текст;
- «Все параметры по умолчанию» - устанавливает шрифт текста как в параметрах карты;
- «Шрифт текста» - установить шрифт текста;
- «Цвет фона» - цвет заливки;
- «Заполнение» - кисть, которой осуществляется заливка фона текста;
- «Толщина границы» - толщина рамки черного цвета вокруг текста. Если 0 – нет рамки;
- «Ширина» - ширина объекта в точках;
- «Высота» - высота объекта в точках;
- «Авторазмеры» - ширину и высоту объекта подбирать автоматически, в зависимости от размера текста;



Прямоугольник и эллипс имеют следующие свойства:

- «Всплывающая подсказка» – текстовая строка. Возможны специальные подстановки аналогичные всплывающей подсказке в параметрах карты;
- «Тип фигуры» - эллипс или квадрат;
- «Цвет фона» - цвет заливки;
- «Заполнение» - кисть, которой осуществляется заливка фона текста;
- «Толщина границы» - толщина рамки черного цвета вокруг текста. Если 0 – нет рамки;
- «Ширина» - ширина объекта в точках;
- «Высота» - высота объекта в точках;



Прогресс-бар имеет следующие свойства:

- «Тип» – горизонтальный (заполнение слева-направо) или вертикальный (заполнение снизу-вверх);
- «Ширина» - ширина объекта в точках;
- «Высота» - высота объекта в точках;
- «Минимум» - минимальное значение канала (прогресс не заполнен);
- «Максимум» - максимальное значение канала (прогресс заполнен полностью);
- «Цвет фона» - цвет фона прогресс-бара;
- «Цвет заполнения» - цвет заполнения прогресс-бара;
- «Сработка» - при достижении этого значения заполнение становится желтым;
- «Тревога» - при достижении этого значения заполнение становится красным;
- «Использовать значения сработки и тревоги» - задействовать два вышеуказанных параметра;



Температура воздуха имеет следующие свойства:

- «Текст» – отображаемый на экране текст с подстановкой (%VALUE это значение канала, т.е. текущая температура);
- «Все параметры по умолчанию» - устанавливает шрифт текста как в параметрах карты;
- «Шрифт текста» - установить шрифт текста;
- «Толщина границы» - толщина рамки черного цвета вокруг текста. Если 0 – нет рамки;
- «Ширина» - ширина объекта в точках;
- «Высота» - высота объекта в точках;
- «Авторазмеры» - ширину и высоту объекта подбирать автоматически, в зависимости от размера текста;



График имеет следующие свойства:

- «Ширина» - ширина объекта в точках;
- «Высота» - высота объекта в точках;
- «Минимум по Y» - минимальное значение, отображаемое по оси Y;
- «Максимум по Y» - максимальное значение, отображаемое по оси Y;
- «Шаг по X, пикселей» - один шаг графика по оси X в точках;
- «Шаг по X, секунд» - количество секунд, через которое график сдвинется влево на «Шаг по X, пикселей»;
- «Толщина графика» - измеряется в точках;
- «Рисовать сетку» - надо рисовать сетку ?
- «Шаг сетки по Y» - шаг сетки по Y в единицах отображаемой величины;
- «Шаг сетки по X» - шаг сетки по X в точках;
- «Отмечать значения кружком» - изменение значения канала будет отмечено кружком;


## Динамический объект на бейсике

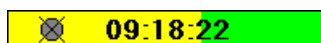
Динамический объект карты может отображать текст, горизонтальную заливку (прогресс бар) и картинку. Его особенность заключается в том, что все свойства задаются в виде выражений на встроенном бейсике.

В APM LanMon 3 появился новый объект «Динамический объект на скрипте». Он отличается от динамического объекта на бейсике следующими моментами:

- Увеличена скорость работы
- Расширена функциональность
- Выражения можно писать на одном из 4х синтаксисов скрипта

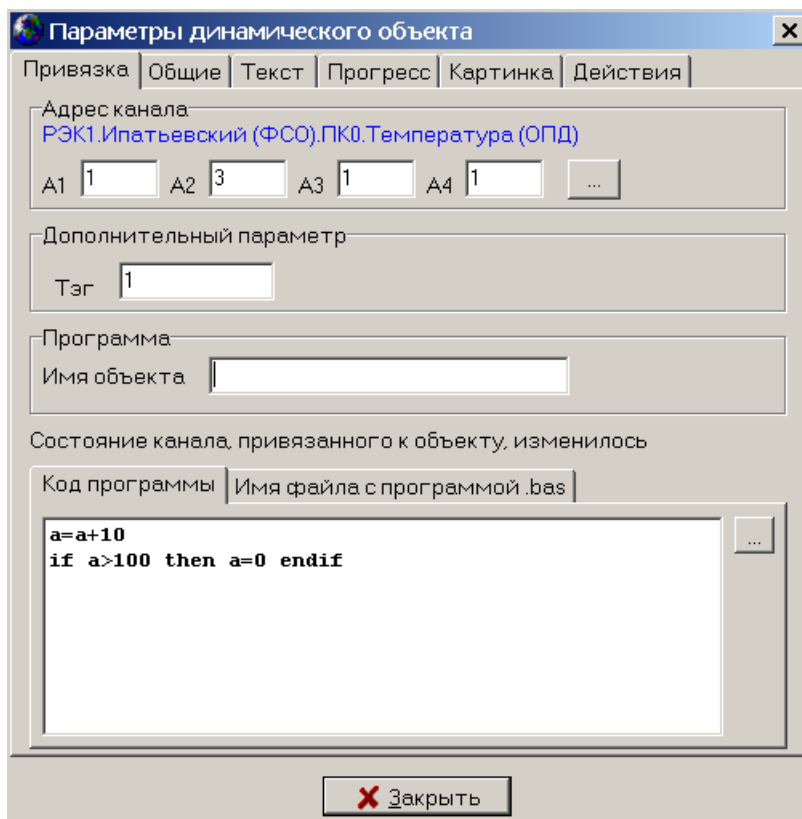
Динамический объект на бейсике считается устаревшим и поддерживается для совместимости. Настоятельно рекомендуется использовать динамический объект на скрипте. Он описан в следующей главе.

Для добавления динамического объекта на бейсике нажмите кнопку  панели инструментов и укажите места на карте для добавления. Пример динамического объекта показан на следующем рисунке:



Приведенный объект отображает текущее время, заливку желтым цветом, и картинку из библиотеки в левой части.

Для открытия окна параметров динамического объекта дважды щелкните мышью на объекте. Откроется окно следующего вида:



Вкладка «Привязка»:

«Адрес канала» - позволяет привязать канал LanMon к динамическому объекту. Такая привязка делает возможным выполнение программы по изменению состоянию объекта. Параметры

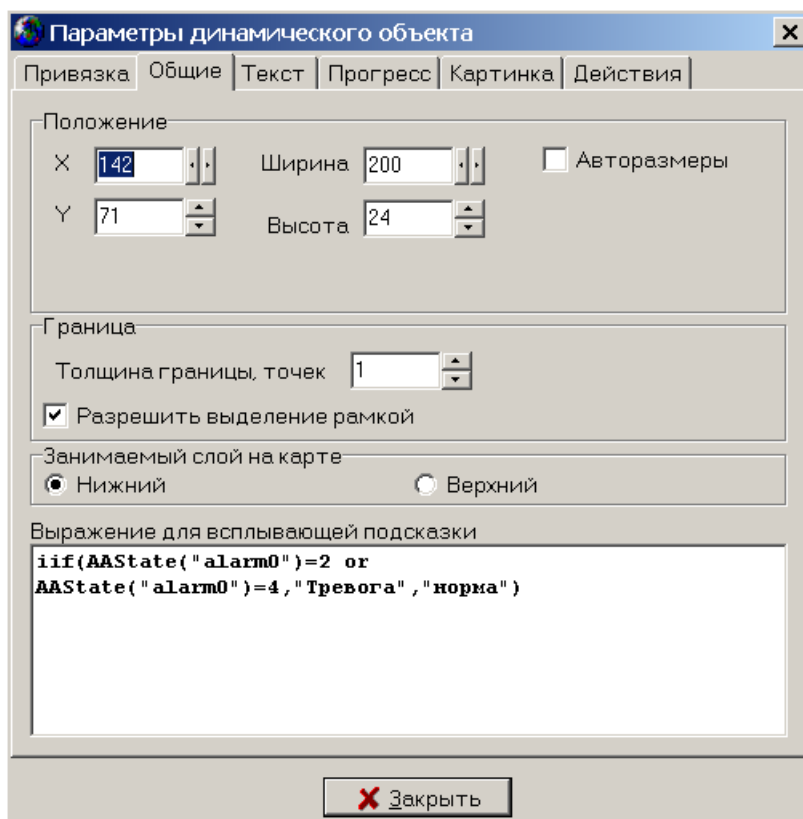
привязанного канала доступны из программ @obj.a1, @obj.a2, @obj.a3, @obj.a4, @obj.state, @obj.value и другие свойства.

«Дополнительный параметр» - целочисленное значение, привязанное к динамическому объекту. Используется по усмотрению программиста. Он доступен из программы на чтение и запись как @obj.tag типа integer.

«Имя объекта» - имя доступное из внешних программ на бейсике. В локальных программах удобнее использовать ссылку на локальный объект @obj

«Программа на изменение состояния канала» - задается код программы или имя файла с программой. Выполняется при изменении состояния канала с адресом «Адрес канала». Если адрес канала не задан то эта программа выполняться не будет.

Вкладка «Общие»



В разделе «Положение» задается положение объекта на карте. X, Y отсчитываются относительно центра объекта.

«Авторазмеры» - если установлена эта галочка – ширина и высота объекта будут выбираться автоматически, независимо от установленных значений.

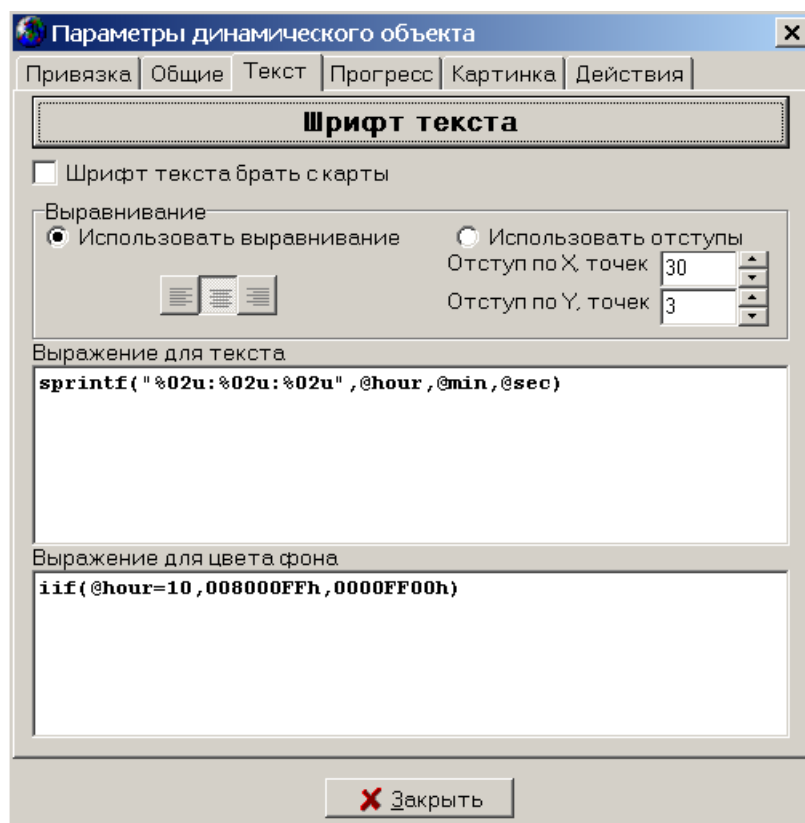
«Толщина границы» - задается толщина линии по контуру объекта в экранных точках. Обводка объекта всегда черного цвета.

«Разрешить выделение рамкой» - при наведении курсора мыши на объект, он обводится рамкой. Действует только в режиме выполнения проекта. Обычно используется если объект реализует действия на щелчки мышью.

«Занимаемый слой на карте» - объект может занимать один из слоев на карте: верхний или нижний.

«Выражение для всплывающей подсказки» - выражение на бейсике формирует текстовую строку для всплывающей подсказки. Рекомендации: удобно пользоваться функциями iif() и sprintf(). В приведенном примере функцией AAState() анализируется состояние аналогового аларма с названием «alarm0» и выдается строка «Тревога» или «норма».

Вкладка «Текст» - задает параметры отображения текста и цвета фона



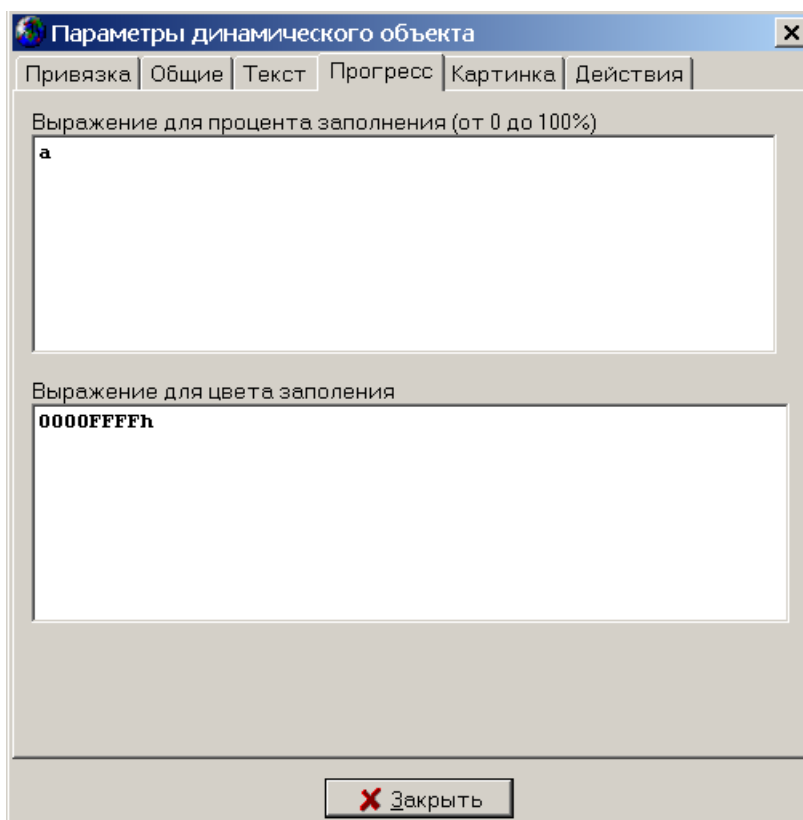
«Шрифт текста брать с карты» - установить шрифт текста объекта как в параметрах карты.

Выравнивание текста предусматривает два режима: выравнивание по горизонтали или отступы от верхнего левого угла объекта в экранных точках.

Выражение для текста. В приведенном примере производится вывод текущего времени на экран. Время форматируется функцией `sprintf()`. Если Вы не планируете вывод текста оставьте выражение пустым.

Выражение для цвета фона. Должно иметь целочисленное значение – цвет фона для динамического объекта. Чтобы вставить в редактор программы константу для цвета нажмите правую клавишу мыши и в появившемся меню выберите «вставить цвет». В приведенном примере в 10 часов объект имеет один цвет в остальное время другой.

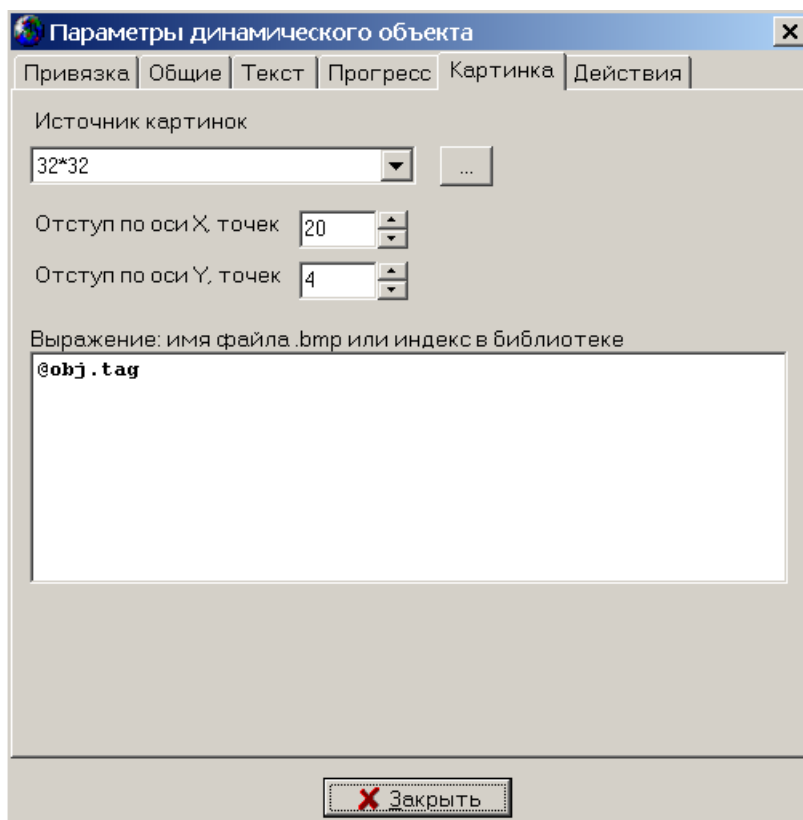
Вкладка «Прогресс» - позволяет отображать заливку слева-направо (прогресс-бар).



Выражение для процента заполнения должно иметь целочисленное значение от 0 до 100. В приведенном примере значение заливки берется из переменной "a".

Для применения фиксированного цвета заполнения просто введите константу цвета.

Вкладка «Картинка» - позволяет отображать картинку из библиотеки APM LanMon или с диска из файла .bmp



«Источник картинок» - выбор библиотеки картинок или «Из файла bmp».



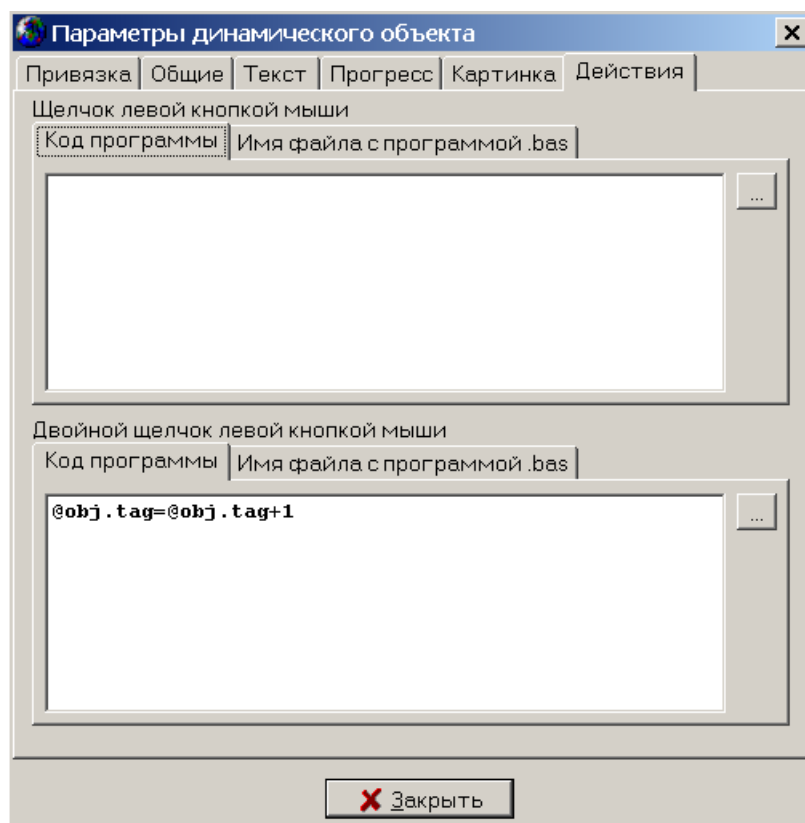
Кнопка «...» служит для просмотра выбранной библиотеки картинок.

«Отступы» - отступы левого верхнего угла картинке от левого верхнего угла объекта.

«Выражение» - определяет какую картинку отображать. Значение выражения – целочисленный индекс картинки в выбранной библиотеке, начиная с нуля или имя файла картинки .bmp. Если имя файла .bmp дано без пути – подразумевается что картинка находится в стандартной поддиректории для картинок .\bmp\

Чтобы не отображать картинку оставьте выражение пустым.

Вкладка «Действия» позволяет задать программу обработчик события на щелчок мышью на объекте.



«Щелчок левой кнопкой мыши» - программа вызывается при одинарном щелчке левой клавишей мыши на объекте в режиме выполнения проекта.

«Двойной щелчок левой кнопкой мыши» - программа вызывается при двойном щелчке левой клавишей мыши на объекте в режиме выполнения проекта.

### Примечания:

Во всех программах динамического объекта доступны ссылки:

@map – ссылается на объект «карта», на которой расположен данный объект;

@obj – ссылается на сам динамический объект;


Во всех окнах ввода программы по нажатию правой клавиши мыши появляется меню с дополнительными возможностями.

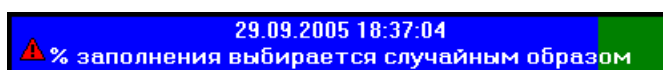
Все программы динамического объекта вычисляются 1 раз в секунду. Кроме программ на вкладке «Действия» и «Привязка».

В LanMon бейсике все переменные глобальные. Во всех выражениях можно использовать переменные, созданные в других программах (и наоборот).

## Динамический объект на скриптовых языках

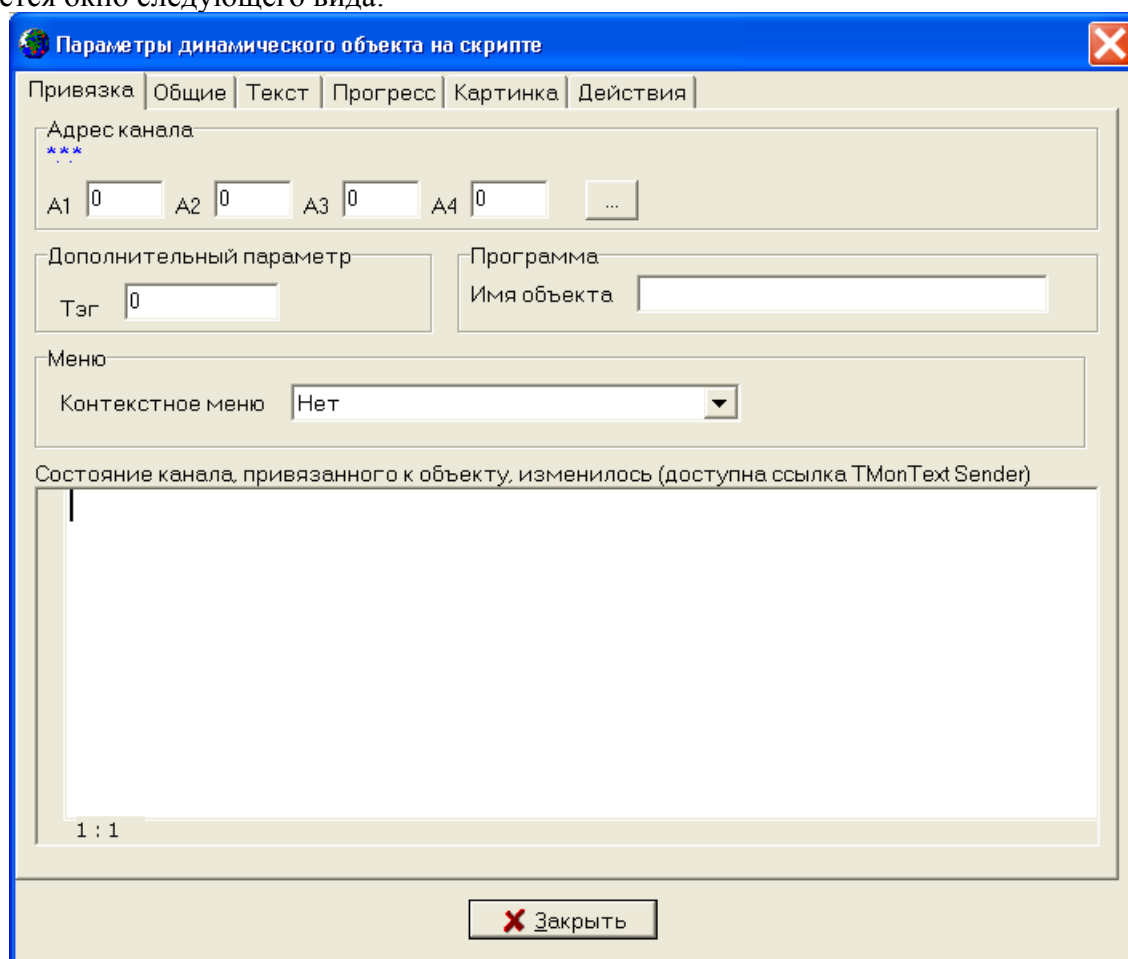
Динамический объект на скрипте может отображать текст, горизонтальную заливку (прогресс бар) и картинку. Его особенность заключается в том, что все свойства задаются в виде выражений на одном из 4-х скриптовых языков. Тип языка для выражений динамического объекта задается в параметрах карты и действует на все ее объекты. Проще говоря, все объекты на одной карты должны использовать выражения на одном языке. В выражениях на скрипте можно вызывать любые стандартные функции, а также функции пользователя из основной программы проекта. Размер кода в свойствах ограничен 500 символами.

Для добавления динамического объекта на бейсике нажмите кнопку  панели инструментов и укажите места на карте для добавления. Пример отображения динамического объекта на карте показан на следующем рисунке:



Приведенный объект отображает текущее дату и время, заливку синим цветом, и картинку из библиотеки картинок в левой части. Заливка выполняется слева направо, причем степень заливки выбирается случайным образом и обновляется каждую секунду. Все выражения примера на C++Script. Этот пример можно увидеть в проекте «Пример проекта на C++». Надо помнить: динамические объекты корректно прорисовываются только в режиме выполнения проекта ! Все свойства объекта пересчитываются один раз в секунду.

Для открытия окна параметров динамического объекта дважды щелкните мышью на объекте. Откроется окно следующего вида:



### Вкладка «Привязка»:

«**Адрес канала**» - позволяет привязать канал LanMon к динамическому объекту. Такая привязка делает возможным выполнение программы по изменению состоянию объекта. Код этой программы на скрипте можно писать на этой же вкладке. Параметры привязанного канала доступны из программ Sender.A1, Sender.A2, Sender.A3, Sender.A4, Sender.STATE, Sender.VALUE и другие свойства.

«**Дополнительный параметр**» - целочисленное значение, привязанное к динамическому объекту. Сохраняется в файле карты. Может использоваться по усмотрению программиста. Он доступен из программы на чтение и запись как Sender.Tag типа integer.

«**Имя объекта**» - имя доступное из внешних программ на бейсике. В локальных программах удобнее использовать ссылку на локальный объект Sender.

«**Программа на изменение состояния канала**» - задается код программы на скрипте. Выполняется при изменении состояния канала с адресом «Адрес канала». Если адрес канала не задан, то эта программа выполняться не будет. При написании программы можно использовать контекстное меню с дополнительными возможностями редактирования.

### Вкладка «Общие»

Параметры динамического объекта на скрипте

Привязка | **Общие** | Текст | Прогресс | Картинка | Действия

Положение

X: 228    Ширина: 420    ☐ Авторазмеры

Y: 209    Высота: 44    ☒ Многострочный

Граница

Толщина границы, точек: 4

☒ Разрешить выделение рамкой

Занимаемый слой на карте

☒ Нижний    ☐ Верхний

Выражение для всплывающей подсказки (доступна ссылка TMonText Sender)

"Подсказка строка 1\\nПодсказка строка 2"

1 : 1

☒ Закрыть

В разделе «**Положение**» задается положение объекта на карте. X, Y отсчитываются относительно центра объекта.

«**Авторазмеры**» - если установлена эта галочка – ширина и высота объекта будут выбираться автоматически, независимо от установленных значений ширины и высоты.

«**Многострочный**» - использовать многострочный текст. Перенос по словам делается автоматически. Принудительный перенос это символы «\n».

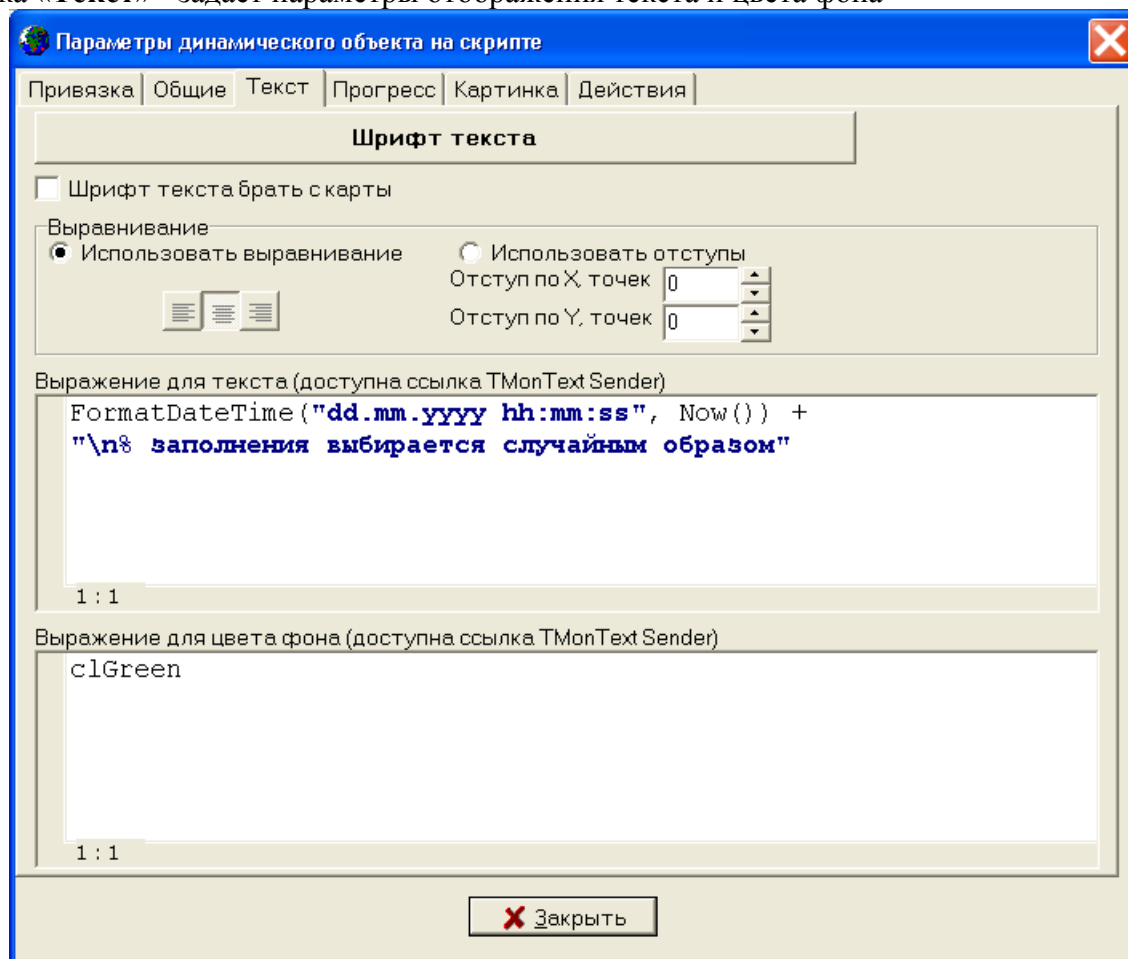
«**Толщина границы**» - задается толщина линии по контуру объекта в экранных точках. Граница объекта всегда черного цвета.

«**Разрешить выделение рамкой**» - при наведении курсора мыши на объект, он обводится рамкой. Действует только в режиме выполнения проекта. Обычно используется, если объект реализует действия на щелчки мышью.

«**Занимаемый слой на карте**» - объект может занимать один из слоев на карте: верхний или нижний.

«**Выражение для всплывающей подсказки**» - выражение на скрипте формирует текстовую строку для всплывающей подсказки. Рекомендации: удобно пользоваться функциями `Imiif()` и `sprintf()`. В приведенном примере выражение задано в виде строковой константы.

Вкладка «**Текст**» - задает параметры отображения текста и цвета фона



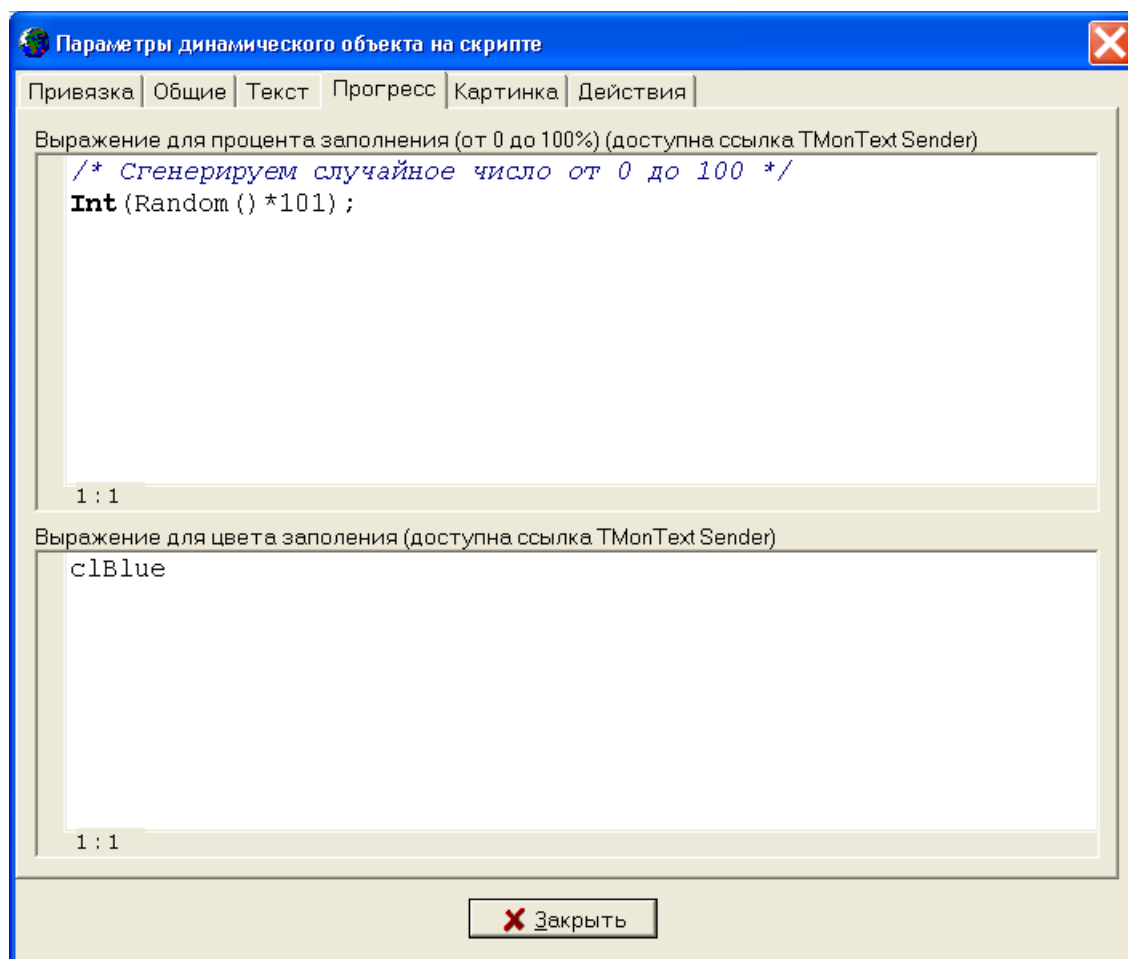
«**Шрифт текста брать с карты**» - установить шрифт текста объекта как в параметрах карты.

«**Выравнивание**» - выравнивание текста предусматривает два режима: указанное выравнивание по горизонтали или отступы от верхнего левого угла объекта верхнего левого угла текста в экранных точках.

«**Выражение для текста**» В приведенном примере производится вывод текущего времени на экран. Время форматируется функцией `FormatDateTime()`. Ко времени прибавляется текстовая строка.

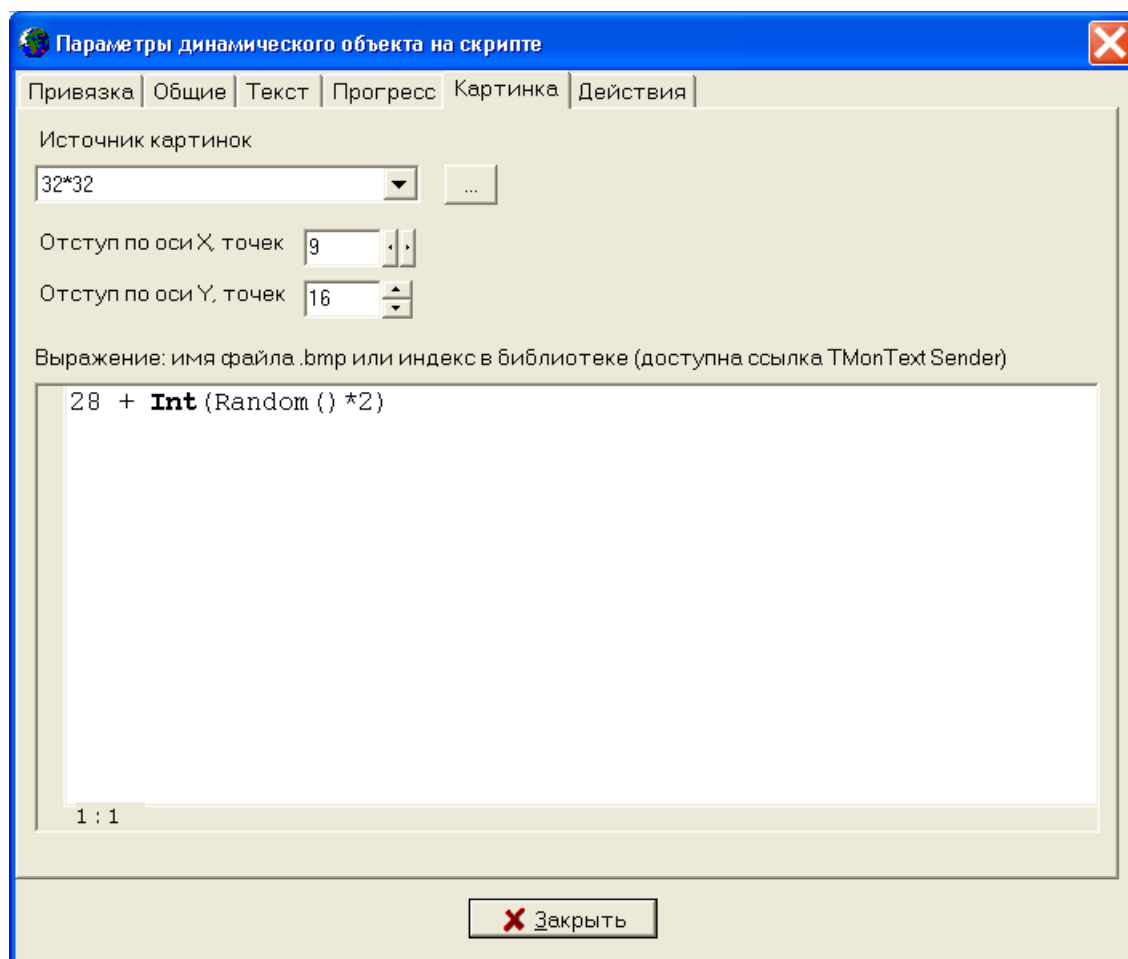
«**Выражение для цвета фона**» Должно иметь целочисленное значение – цвет фона в формате RGB для динамического объекта. Чтобы вставить в редактор программы константу для цвета нажмите правую клавишу мыши и в появившемся меню выберите «Вставить константу цвета символьную...». В приведенном примере используется константа для зеленого цвета.

Вкладка «**Прогресс**» - позволяет отображать заливку слева-направо (прогресс-бар).



Выражение для процента заполнения должно иметь целочисленное значение от 0 до 100. В приведенном примере значение заливки генерируется случайным образом. Для применения фиксированного цвета заполнения просто введите константу цвета.

Вкладка «**Картинка**» - позволяет отображать картинку из библиотеки или с диска из файла в формате bmp



«**Источник картинок**» - выбор библиотеки картинок APM LanMon или «Из файла bmp». Кнопка «...» служит для просмотра выбранной библиотеки картинок или для вставки имени файла.

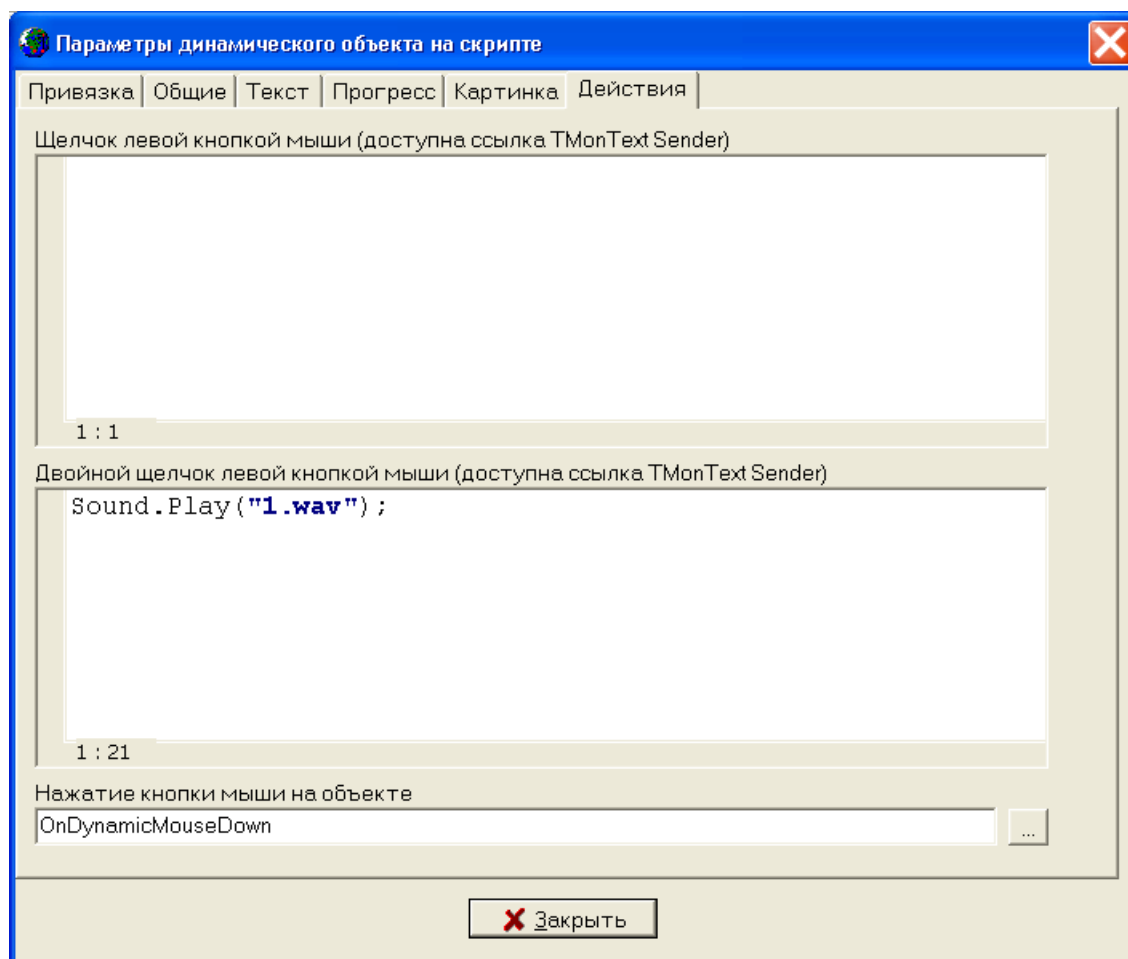
«**Отступы**» - отступы левого верхнего угла картинки от левого верхнего угла объекта.

«**Выражение**» - определяет какую картинку отображать. Значение выражения – целочисленный индекс картинки в выбранной библиотеки, начиная с нуля или строка - имя файла картинки .bmp. Если имя файла .bmp дано без пути – подразумевается что картинка находится в стандартной поддиректории для картинок проекта .\bmp\

Чтобы не отображать картинку вообще оставьте выражение пустым.

В приведенном примере выбирается картинка из библиотеки «32\*32» по индексу. Индекс формируется случайным образом: 28 или 29.

Вкладка «**Действия**» позволяет задать программы обработчики событий на щелчок мышью на объекте.



«Щелчок левой кнопкой мыши» - программа вызывается при одинарном щелчке левой клавишей мыши на объекте в режиме выполнения проекта.

«Двойной щелчок левой кнопкой мыши» - программа вызывается при двойном щелчке левой клавишей мыши на объекте в режиме выполнения проекта.

В этих обработчиках надо писать не выражение на скрипте, а код. Их этого кода можно вызывать любые функции основного скрипта проекта. Размер кода ограничен 500 символами. Должна быть определена обработка одного из этих событий. Так как перед вызовом обработчика двойного щелчка всегда будет вызываться обработчик одинарного.

«Нажатие кнопки мыши на объекте» - Может быть определен обработчик нажатия кнопки мыши в основной программе. Это позволяет кроме факта нажатия кнопки мыши получить дополнительную информацию: какая именно кнопка мыши нажата, координаты курсора мыши в пределах объекта, какая клавиша-переключатель была нажата в этот момент на клавиатуре. Пример обработчика:

```
// Это обработчик события на нажатие клавиши мыши на объекте карты
void OnDynamicMouseDown(TMonText Sender, TMouseButton Button, TShiftState Shift, int X, int Y)
{
    String b;
    if( Button==mbLeft ) b="левая";
    else if( Button==mbMiddle ) b="средняя";
    else if( Button==mbRight ) b="правая";

    String k="";
    // && - побитовое AND (аналог & в C++)
    if( Shift && ssAlt ) k += " Alt";
    if( Shift && ssCtrl ) k += " Ctrl";
    if( Shift && ssShift ) k += " Shift";
}
```

```
//
DynamicText.Caption = "X="+IntToStr(X) + " Y="+IntToStr(Y) + " "+b + k;
}
```

### Примечания:

Во всех программах динамического объекта доступны ссылки на объекты:

Sender.map – ссылается на объект карта класса TMap, на которой расположен данный объект;

Sender – ссылается на сам динамический объект класса TMonText;


Во всех окнах ввода программ по нажатию правой клавиши мыши появляется меню с дополнительными возможностями.

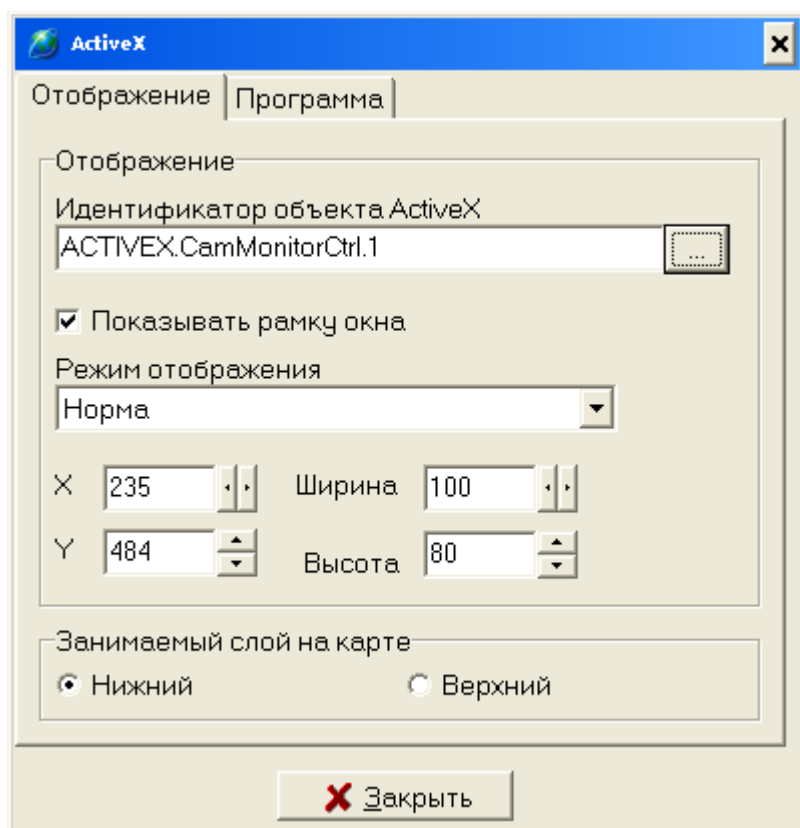
Все программы динамического объекта вычисляются 1 раз в секунду. Кроме программ на вкладках «Действия» и «Привязка».


Во всех выражениях можно использовать глобальные переменные, определенные в основной программе.

## ActiveX объекты

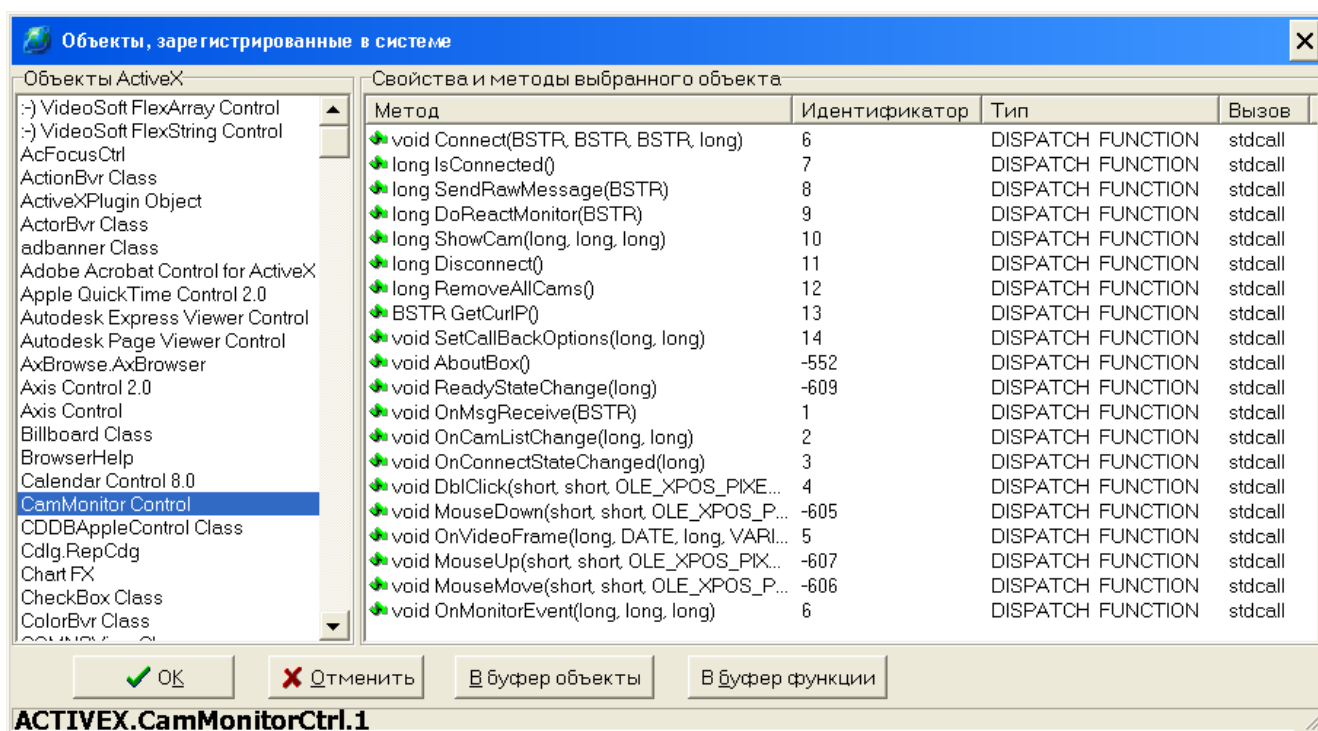
На карте можно разместить любой ActiveX объект, зарегистрированный в системе.

Поместите на карту объект ActiveX. Для этого служит кнопка  панели инструментов. Дважды щелкните мышью на объекте карты – откроется окно свойств ActiveX объекта:



Идентификатор объекта ActiveX это тестовая строка, идентифицирующая объект. Для выбора идентификатора объекта из списка нажмите кнопку . При этом, появится окно следующего вида:

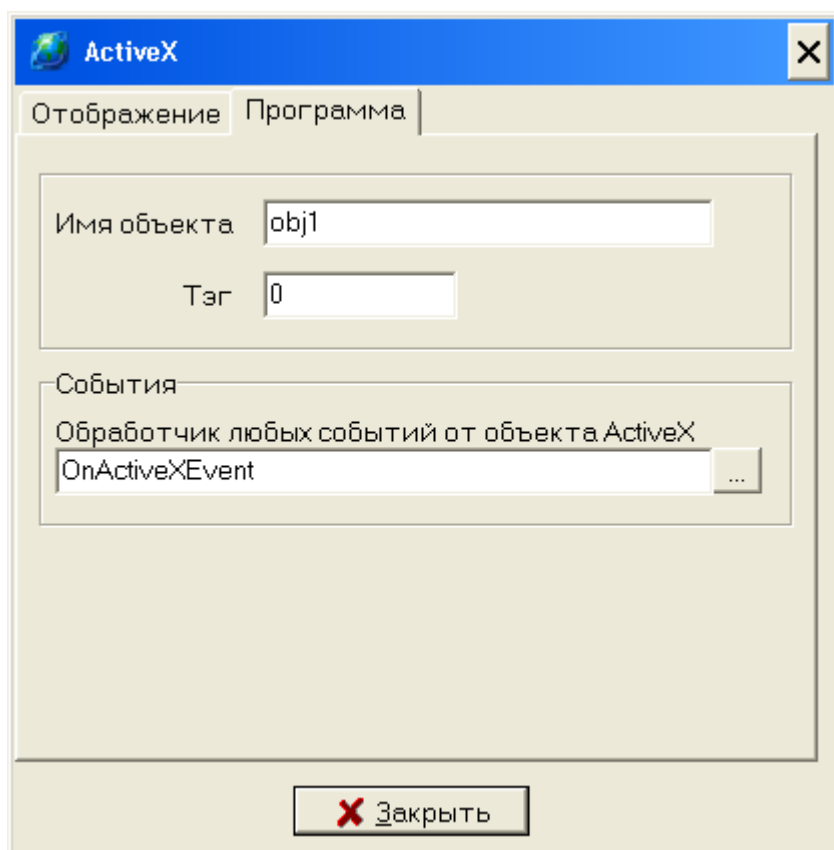




В левой части окна расположен список всех ActiveX объектов, зарегистрированных в системе. Объекты отсортированы в алфавитном порядке. При выделении объекта в списке в правой части окна выводится список методов и свойств этого объекта. В колонке «Идентификатор» выводится уникальный номер для каждого метода или свойства. Этот номер используется для определения типа события в обработчике событий от объекта ActiveX.


В строке статуса печатается идентификатор выбранного ActiveX объекта. Для выбора выделенного объекта нажмите кнопку «ОК». Для отмены выбора нажмите «Отменить».

На вкладке «Программа» задайте «Имя объекта» и название функции – обработчика событий от данного объекта:



Указанное имя объекта становится доступным в программе на скрипте. Через имя можно вызывать функции и свойства объекта. Следующие функции доступны для вызовы из программы:

Описание	Комментарии
function Call(Name:String; Type:String; Params: Variant): Variant	<p>Выполнить функцию или вызвать свойство объекта ActiveX.</p> <p><i>Name</i> – Имя функции или свойства.</p> <p><i>Type</i> – Строка, обозначающая тип вызываемого объекта. Может быть одним из следующих:</p> <p>"FUNC" – <i>Name</i> это имя функции</p> <p>"PROPERTYGET" - <i>Name</i> это имя свойства, возвращающего значение</p> <p>"PROPERTYPUT" - <i>Name</i> это имя свойства, принимающего значение</p> <p><i>Params</i> – массив входных параметров для вызова функции.</p> <p>Функция <i>Call</i> возвращает значение, которое возвращает вызванный метод или свойство объекта ActiveX.</p>

Для добавления в текст программы обработчика событий от данного объекта нажмите кнопку . Приведем пример функции – обработчика событий на C++ скрипте:

```
/*
  Sender - ссылка на объект карты TMonActiveX, от которого пришло событие
  MessageType - идентификатор события (можно посмотреть в списке функций при
                добавлении ActiveX на карту).
  Params - массив параметров события
*/
```

```
void OnActiveXEvent(TMonActiveX Sender, int MessageType, Variant & Params)  
{  
}
```

В проекте «Пример работы с CamControl» иллюстрируется вызов функций ActiveX объекта и обработка сообщений.

## Меню карты

У карты можно определить пользовательское меню, доступное в режиме выполнения проекта. Допускается создание до пяти разных меню. Каждое из меню можно привязать к одной или нескольким картам.

Меню описываются в текстовых файлах. Файлы с описанием меню называются `fmenuNN.dat` и хранятся в поддиректории `.menu\` проекта. NN в названии файла это номер меню от 00 до 04. Для создания меню карты загрузите проект в режиме редактирования и выберите в меню «Ресурсы / Меню карты / Меню 1». Появится окно с редактором текстового файла описателя меню. Также, вы можете отредактировать описание меню любым текстовым редактором.

Далее приведен пример файла `fmenu00.dat`:

```
; Это файл описания меню карты APM LanMon.
; Первая строка содержит название данного меню в двойных кавычках.
; Далее идет перечисление пунктов меню в формате:
; "Название пункта меню"
; "Название пункта меню", "Название функции скрипта - обработчика"
; Если задано название функции обработчика не задано - считается что
; это пункт строки меню.
; Сепаратор пунктов меню задается так:
; "_"
; Обработчик событий от меню выглядит так:
; void OnMapMenuItem(TMap Sender, int MenuIndex) {}
```

"Меню 1"

"Пункт 1"

"Пункт 1.1", "OnMapMenuItem"

"\_"

"Пункт 1.2", "OnMapMenuItem"

"Пункт 2"

"Пункт 2.1", "OnMapMenuItem"

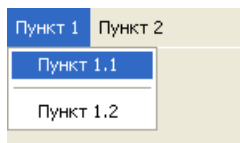
"Пункт 2.2", "OnMapMenuItem"

Строка после символа «;» считается комментарием и игнорируется. Все строковые литералы заключаются в кавычки (название пункта меню или функции-обработчика). Первой строкой в файле идет название меню *"Меню 1"*. Название меню используется только на этапе редактирования проекта. После названия меню идет название первого пункта строки меню. Далее идет список пунктов всплывающего меню с названием функции обработчика событий. Как только встречается пункт без указания функции-обработчика, считается, что это название следующего пункта строки меню.

После создания меню надо «привязать» его к одной или нескольким картам. Для этого в режиме редактирования проекта, в меню карты выберите «Меню / Параметры карты...». В появившемся окне на вкладке «Общие» выберите название меню из всплывающего списка «Основное меню». После выбора меню нажмите кнопку «ОК».

В программе на скрипте должна быть определена функция `void OnMapMenuItem(TMap Sender, int MenuIndex) {}`. Эта функция получает управление при выборе одного из пунктов меню. Параметр *Sender* – ссылка на объект карты, на котором расположено меню. Параметр *MenuIndex* номер пункта меню с нуля. Пункты меню нумеруются в порядке, в котором они перечислены в файле описания меню. Для добавления в текст программы указанного обработчика, откройте редактор программ и в меню выберите «События / Обработчик событий от меню карты».

Меню карты появляется только в режиме выполнения проекта. В режиме редактирования у карты predetermined меню. Приведенное в примере описание меню в режиме выполнения проекта выглядит так:



Пример меню приведен в демонстрационном проекте «Пример проекта LanMon C++». Этот проект устанавливается в комплекте с APM LanMon.

### Список отображаемых объектов на картах

Вы можете посмотреть список отображаемых объектов на всех загруженных картах в табличном виде. Для открытия списка выберите в меню пункт «Просмотр Список \ объектов...». Появится окно следующего вида:

Список отображаемых объектов на картах

Операции

Формирование названия канала

☐ A1 ☐ A2 ☒ A3 ☒ A4

Фильтр

По типу объекта: Все типы объектов

По подстроке в названии:

По типу данных: По всем каналам

По адресу канала:

По карте: По всем картам

☐ По наличию привязки к адресу канала

☐ По отсутствию привязки к адресу канала

№	Тип объекта	Название	Адрес	Тип д	Имя в	Меню	Алармы	Связи	Карта	X Y
17	TMonStd	Дом 21 п.3.Лифт	1.2.3.1	24		Меню 1			Диспетчеризация	84, 824
18	TMonStd	Дом 28 п.1.Лифт	1.1.13.1	24		Меню 1			Диспетчеризация	251, 545
19	TMonStd	Дом 13 п.1.Лифт	1.1.1.1	24		Меню 1			Диспетчеризация	58, 73
20	TMonStd	Дом 13 п.1.Вход в машинное	1.1.1.2	11		Меню 1	Вход в МП		Диспетчеризация	118, 79
21	TMonText	Дом 13 п.1.последний код	1.1.1.10	7		Меню 1			Диспетчеризация	242, 60
22	TMonStd	Дом 13 п.2.Лифт	1.1.2.1	24		Меню 1			Диспетчеризация	59, 150
23	TMonStd	Дом 13 п.2.Вход в машинное	1.1.2.2	11		Меню 1	Вход в МП		Диспетчеризация	118, 152
24	TMonText	Дом 13 п.2.последний код	1.1.2.10	7		Меню 1			Диспетчеризация	243, 144
25	TMonStd	Дом 13 п.3.Лифт	1.1.3.1	24		Меню 1			Диспетчеризация	58, 228
26	TMonStd	Дом 13 п.3.Вход в машинное	1.1.3.2	11		Меню 1	Вход в МП		Диспетчеризация	118, 228

Список отображаемых объектов содержит следующие колонки:

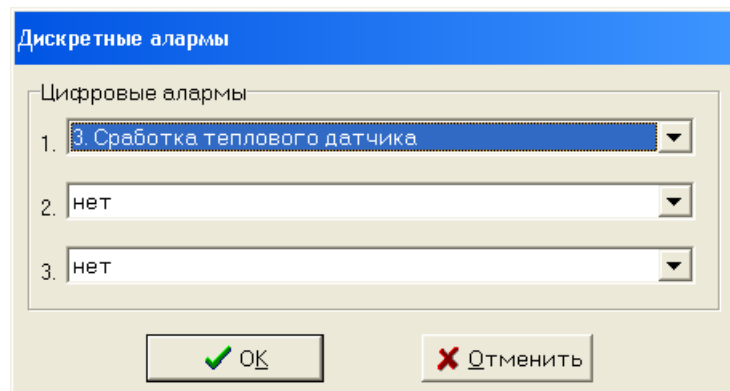
- № - номер объекта с единицы
- Тип объекта – тип объекта
- Название – название канала, привязанного к объекту
- Адрес – адрес канала
- Тип данных – номер типа данных канала
- Имя в скрипте – имя объекта, доступное из программы
- Меню – всплывающее меню, привязанное к данному объекту
- Алармы – список дискретных алармов объекта
- Связи – список связей
- Карта – название карты, на которой лежит объект
- X, Y – координаты объекта на карте

В разделе «Формирование названия канала» задается способ формирования колонки «Название» галочками «A1» «A2» «A3» «A4». В разделе «Фильтр» задаются условия фильтрации списка объектов.

Для показа выбранного объекта на карте выберите в меню «Операции \ Показать объект на карте» или дважды щелкните мышью на объекте в списке. Будет показана карта с объектом, сам объект будет выделен красной рамкой.

Для изменения настроек (свойств) выбранного объекта выберите в меню «Операции \ Свойства объекта...». Появится окно с настройками объекта.

Для установки указанных алармов по всем объектам выберите в меню «Операции \ Установить алармы по всем объектам...». В появившемся окне выберите дискретные алармы.



После нажатия кнопки «ОК» выбранные алармы будут установлены у всех объектов в списке. Удобно отфильтровать нужные объекты и присвоить им алармы.

Для копирования списка объектов в буфер обмена Windows выберите в меню «Операции \ Копировать содержимое списка в буфер обмена».

Основное назначение данного списка в групповом изменении алармов, а также в контроле правильности назначения алармов, меню, связей и прочее объектам на картах.

## IP телефония по протоколу H.323

В АРМ LanMon интегрирован клиент (терминал) IP телефонии на базе протокола H.323. Протокол H.323 является общепринятым стандартом IP телефонии. Клиент IP телефонии в АРМ LanMon совместим с любыми устройствами, поддерживающими стандарт H.323. Работа IP телефонии тестировалась с Microsoft NetMeeting, с различными клиентами IP телефонии на базе библиотеки OpenH323, с аналоговым коммутатором Cisco ATA 186, со шлюзом IP телефонии sos95gw.

Все настройки клиента IP расположены в настройках проекта на вкладке «Голосовая связь»:

Для включения клиента IP телефонии установите галочку «Использовать голосовую связь в стандарте H.323».

В поле «Алиас» надо задать текстовый идентификатор данного клиента. Он же будет использоваться и при регистрации на гейткипере.

При работе с кодеками GSM или G.711 возможно использование режима экономии сетевого трафика: если в разговорном тракте тишина – данные по сети не передаются. Включением этого режима производится галочкой «Запретить определение тишины для кодеков GSM и G.711». При разговоре с переговорным блоком СОС-95: УИР-РЦ, БГС, БДКЛ и прочие эту галочку необходимо установить.

Для настройки автоматического перенаправления вызова при занятости нашего номера, задайте в соответствующем поле ввода адрес абонента в формате H.323: `[alias@][transport$]host[:port]` При работе с гейткипером можно задать только алиас (или телефонный номер). Если оставить поле пустым – переадресация выполняться не будет.

Для настройки автоматического перенаправления вызова если мы не берем трубку задайте максимальное время ответа и адрес абонента в формате H.323: `[alias@][transport$]host[:port]` Если оставить поле ввода адреса пустым – перенаправление выполняться не будет.

В поле «Дополнительные параметры» можно задать текстовую строку из дополнительных параметров клиента IP телефонии:

–**g711-ulaw** использовать звуковой кодек «СCITT u-Low, 8000 кГц, 8 бит, моно».

Рекомендуется использовать всегда. При работе со звуковым шлюзом sos95gw это единственный вариант.

–**j N** установить размер звукового буфера в миллисекундах, где N – количество миллисекунд. По умолчанию используется 50 мс. На сетях 100 Мбит можно уменьшать буфер до 25 мс. Это уменьшит задержку звука. На медленных каналах связи размер буфера можно увеличивать до 250 мс и более. Эту опцию можно вообще не указывать – все будет нормально работать в большинстве случаев.

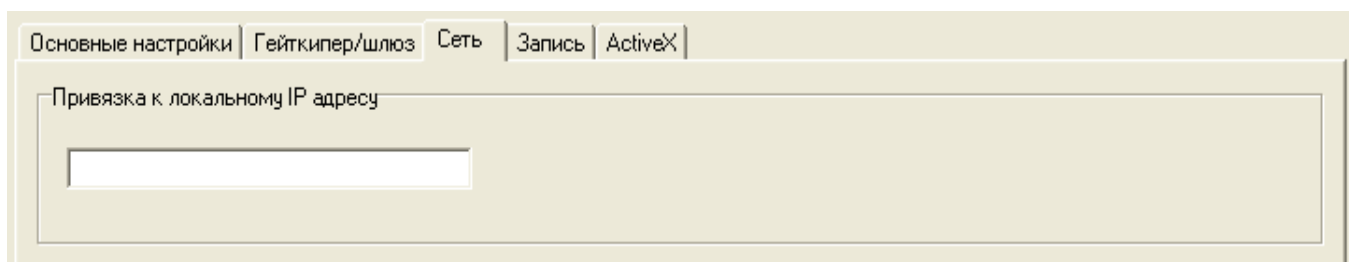
На вкладке «Гейткипер/шлюз» задаются параметры работы с гейткипером:

Для регистрации на гейткипере (привратнике) установите галочку «Использовать гейткипер». Далее необходимо указать IP адрес гейткипера или его идентификатор. Если поле «Адрес» оставить пустым – будет произведен поиск в сети гейткипера с идентификатором, указанным в поле «Идентификатор ID». Если гейткипер требует авторизацию по протоколу H.235 необходимо ввести пароль в поле «Пароль». Если оставить поле «Пароль» пустым – авторизация использоваться не будет. При использовании гейткипера все звонки проходят через него. В этом случае, в записной книжке у абонентов можно указывать лишь алиас (или телефонный номер) абонента без IP адреса.

Если гейткипер не используется, то возможна работа через шлюз H.323. Задайте в поле «Шлюз» IP адрес шлюза. Если IP адрес шлюза не задан – шлюз использоваться не будет. При использовании шлюза все исходящие звонки будут перенаправляться на него. В этом случае, в записной книжке у абонентов можно указывать лишь алиас (или телефонный номер) абонента без IP адреса.

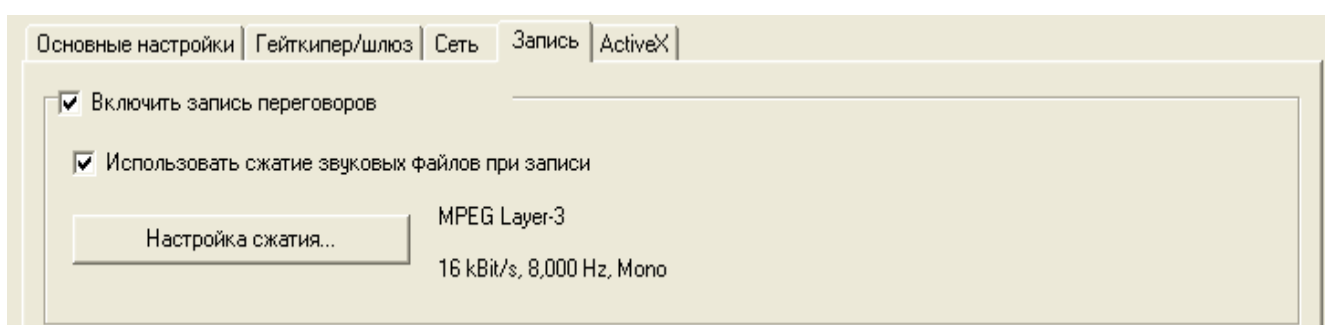
На вкладке «Сеть» задается привязка сервиса IP телефонии к локальным сетевым интерфейсам:





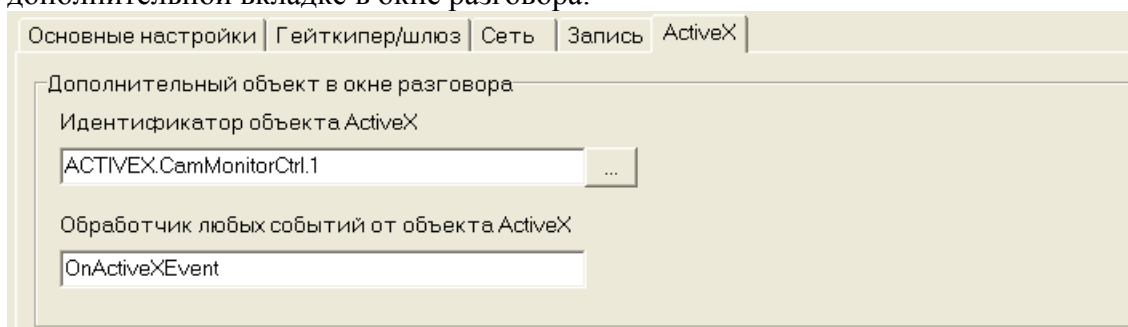
Эта возможность используется, если компьютер имеет более одного IP адреса. Если оставить поле «*Привязка к локальному IP адресу*» пустым, то привязка будет осуществляться ко всем адресам компьютера (интерфейс 0.0.0.0). Введите в это поле IP адрес, к которому нужно осуществить привязку.


На вкладке «*Запись*» включается запись разговоров на диск и настраивается сжатие:



Сжатие записываемых разговоров позволяет экономить дисковое пространство. Для сжатия можно использовать любой из установленных в системе кодеков. При выборе кодека обратите внимание на то, что кодеки не умеют делать изменение частоты (передискретизацию). Т.е. если при разговоре используется кодек **g711-ulaw** с частотой 8000 Гц нельзя пытаться сжать его кодеком с частотой 12000 Гц. На картинке показаны оптимальные параметры сжатия разговора с кодеком **g711-ulaw** в формате MP3. Кодек **g711-ulaw** используется всегда при разговоре с переговорными блоками СОС95.

На вкладке «*ActiveX*» можно произвести выбор объекта ActiveX для отображения на дополнительной вкладке в окне разговора:



Для выбора одного из объектов ActiveX, установленных в системе нажмите кнопку , в появившемся списке выберите объект и нажмите кнопку «ОК». Если поле «*Идентификатор объекта ActiveX*» оставить пустым – дополнительная вкладка в окне разговора будет скрыта. Основное назначение размещения объекта в окне разговора – показ видео (абонента, с которым мы разговариваем). Указанный здесь ActiveX объект доступен из программы по имени «*VoiceObject*». Вы можете вызывать его функции и свойства из программы через вызов метода *Call*. Для обработки событий от указанного ActiveX можно задать имя функции – обработчика событий в программе. Приведем пример функции – обработчика событий на C++ скрипте:

/\*


*Sender* - ссылка на объект карты *TMonActiveX*, от которого пришло событие  
*MessageType* - идентификатор события (можно посмотреть в списке функций при  
 добавлении *ActiveX* на карту).

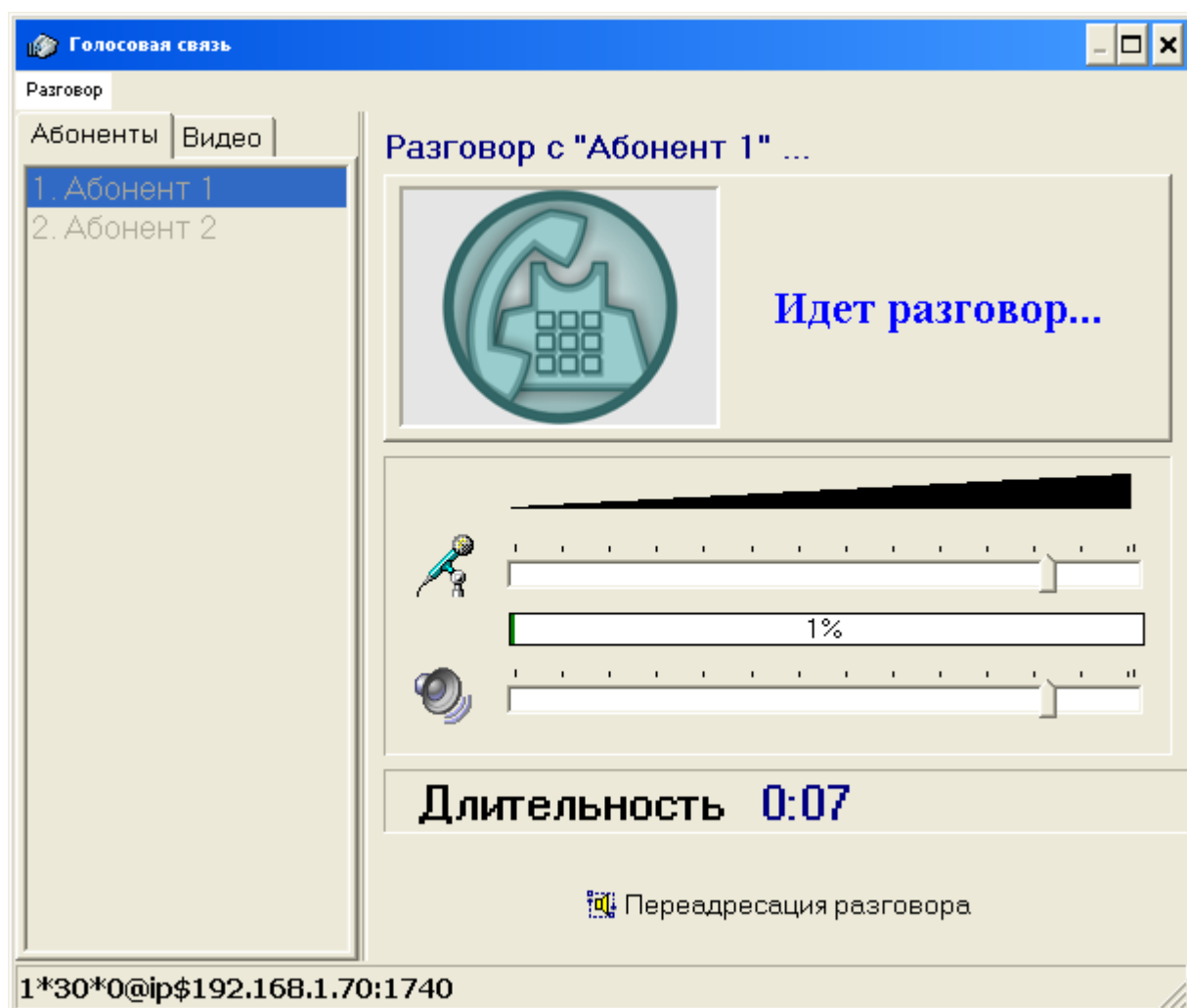
*Params* - массив параметров события

\*/

```
void OnActiveXEvent(TMonActiveX Sender, int MessageType, Variant & Params)
{
}
```

**ВНИМАНИЕ:** после изменения любых параметров на вкладках «Основные настройки», «Гейткипер/шлюз», «Сеть» проект надо перезагрузить (закрыть и открыть заново). Параметры на других вкладках применяется после нажатия клавиши «ОК», без необходимости перезагрузки проекта.

Для открытия окна IP телефонии выберите пункт меню «Звук/Голосовая связь» или нажмите на кнопку . Появится окно голосовой связи:



Пункты меню «Абонент» и «Настройка» служат для настройки и доступны только в режиме редактирования проекта.

В левой части окна расположен список абонентов. Он заполняется на этапе наладки проекта. Для его редактирования используйте меню «Абонент».

Кнопка «Позвонить» с изображением телефона служит для звонка абоненту, выбранному в списке. Для завершения звонка служит эта же кнопка. При поступлении входящего звонка эта же кнопка служит для поднятия трубки (ответа на звонок). Клавиша «Enter» всегда имитирует нажатие этой кнопки.

В секции «Длительность» указана длительность последнего разговора.

Кнопка «Переадресация разговора» позволяет переадресовать вызов любому абоненту из списка. Нажатие на нее доступно только в процессе разговора. После нажатия на нее необходимо указать абонента из списка и нажать кнопку «Переадресовать» или клавишу «Enter». Клавиша с латинской буквой «Т» также выполняет переадресацию разговора.

Для просмотра и прослушивания архива записанных разговоров выберите в меню «Разговор / Просмотр звукозаписей...».

В режиме редактирования проекта список абонентов можно редактировать. При выборе функции добавления или изменения абонента появляется окно:

Поле «Имя абонента» служит для идентификации абонента в списке.

В поле «Номер в формате» введите адрес абонента в установленном формате или алиас (номер телефона) при работе через гейкипер или шлюз.

Для переговорного блока СОС95 необходимо задать дополнительные параметры.

В поля «Параметр 1» и «Параметр 2» секции «Дополнительно» могут быть установлены произвольные текстовые данные, которые доступны из программы на скрипте. В примере «Пример работы с CamControl» туда пишутся адрес видеосервера и номер камеры для каждого из абонентов.

Список абонентов, параметры каждого абонента и номер выбранного абонента доступны из программы на скрипте через объекты классов *TIPAbonent* и *TIPAbonentList*. Подробнее читайте в документе «Руководство по программированию» (script.pdf).

В программе на скрипте доступен ряд функций для работы с IP телефонией. Эти функции начинаются с префикса «H323». Полное описание функций по работе с IP телефонией приведено в документе «Руководство по программированию» (script.pdf).

Например, для автоматического показа окна IP телефонии при появлении входящих звонков надо постоянно выполнять следующий код на C++ скрипте:

```
if ( H323State()==3 ) H323ShowWindow(1);
```

или установить галочку «*Автоматически показывать окно разговора при поступлении входящего звонка*».

## Формат комплексного номера переговорного устройства сети СОС-95 (БГС, УИР-Р, БДКЛ)

Функция вызова H323Call() принимает в качестве параметра полный адрес абонента. Для звонка на переговорное устройство (ПУ) сети СОС-95 используется специальный формат адреса:

**P\*A[\*C]@IP\_ADDRESS:PORT**

**P** – номер последовательного порта, к которому подсоединен контроллер БКД-М. Допустимые значения от 1 до 255 (для Linux /dev/ttyS0 это порт №1);

**A** – адрес переговорного устройства в сети СОС-95. Допустимые значения от 1 до 255;

**C** – номер переговорного блока в устройстве (разговорного канала). Указывается для многоканальных устройств типа лифтовых контроллеров. Допустимые значения от 0 до 3. Необязательный параметр, по умолчанию – 0.

Расшифровка номера переговорного блока для лифтового контроллера:

0-машинное помещение (основной переговорный блок)

1-лифт

2-дополнительное переговорное устройство №1

3-дополнительное переговорное устройство №2

Для БГС указывайте 0.

Когда мы устанавливаем тип абонента «Переговорный блок СОС-95» все заданные параметры автоматически форматируются по вышеописанному правилу.

IP\_ADDRESS – IP адрес или имя узла абонента.

PORT – порт абонента. По умолчанию (если не задан) 1720. Если звонок осуществляется через шлюз sos95gw установленный на том же компьютере, что и APM LanMon необходимо указывать явно порт как 1740 (в настройках шлюза тоже должен быть указан порт 1740 вместо 1720).

При звонке на обычный клиент IP телефонии (другой APM LanMon) достаточно указать IP адрес абонента. Например: H323Call(«192.168.1.100»). При звонке на клиент IP телефонии, имеющий несколько линий (например Cisco ATA 186) перед адресом надо указать алиас линии. Например: H323Call(«line1@192.168.1.100»).

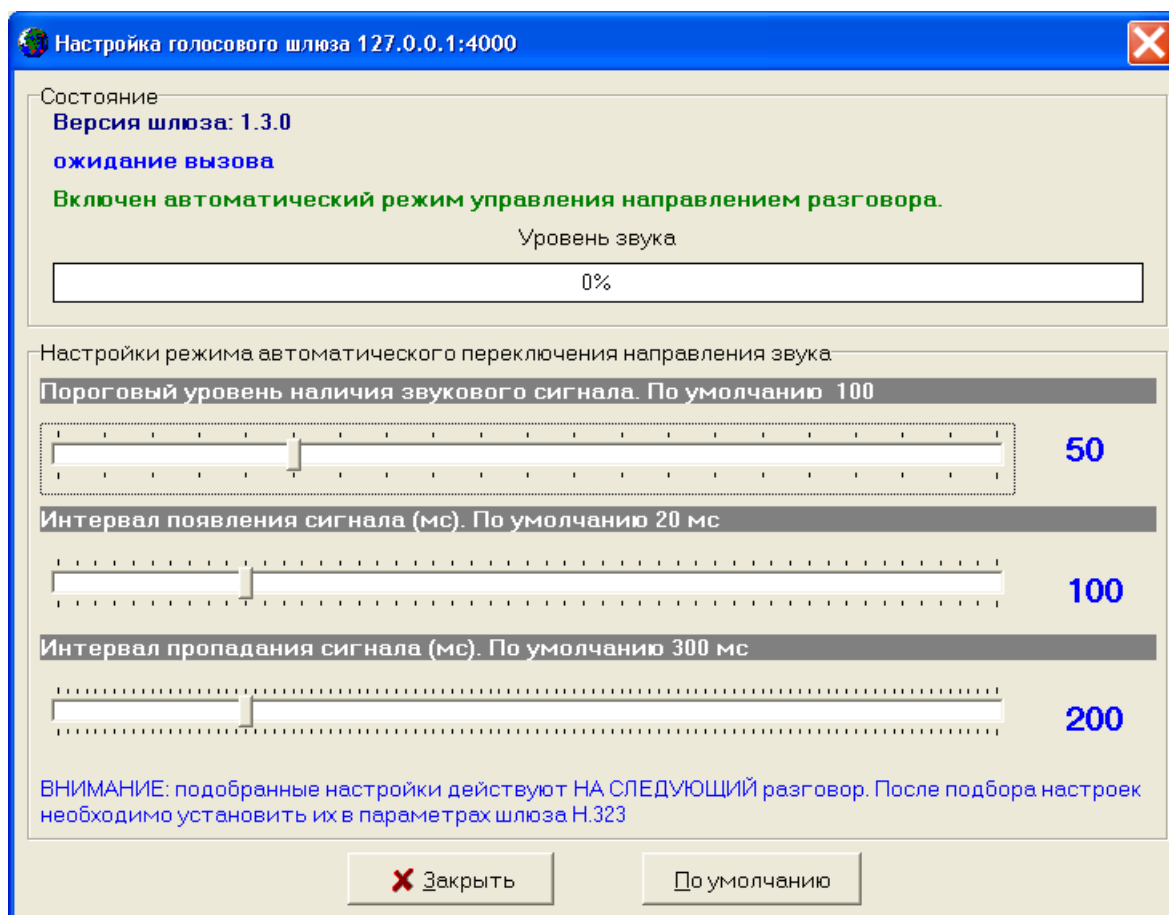
## Особенности разговора с устройствами СОС-95

Разговор с переговорными устройствами сети СОС-95 производится через шлюз sos95gw. Все переговорные устройства в сети СОС-95 являются полудуплексными. Переключение направления разговора осуществляется автоматически от голоса диспетчера или в ручном режиме клавишей пробел. Для работы с переключением направления звука от голоса диспетчера необходимо настроить три параметра:

- Threshold – уровень наличия звукового сигнала.
- Signal Deadband – интервал появления сигнала (мс).
- Silence Deadband – интервал пропадания сигнала (мс).


Эти параметры задаются в текстовом файле настроек шлюза sos95gw. Настройка заключается в подборе значений для обеспечения комфортного переключения направления звука. Для настройки шлюза выберите в меню окна «Голосовая связь» пункт «Связь / Настройка шлюза sos95gw...». IP адрес настраиваемого шлюза берется из настроек текущего абонента. Данное окно служит для подбора параметров, которые надо потом задать в

настройках шлюза. Если этого не сделать, то после перезапуска шлюза они будут утеряны. Окно настроек параметров шлюза показано на следующем рисунке:



Любые изменения параметров в этом окне немедленно устанавливаются в шлюзе, но действуют только на следующий разговор.

### Как настроить микшер

Для открытия окна микшера выберите пункт меню «Звук \ Настройка микшера...» или нажмите кнопку . В APM LanMon есть два профиля настроек микшера: основной и альтернативный. Основной профиль настроек автоматически устанавливается после старта APM LanMon и служит для разговора с лифтами, переговорными блоками, IP - фонами и другими APM LanMon.

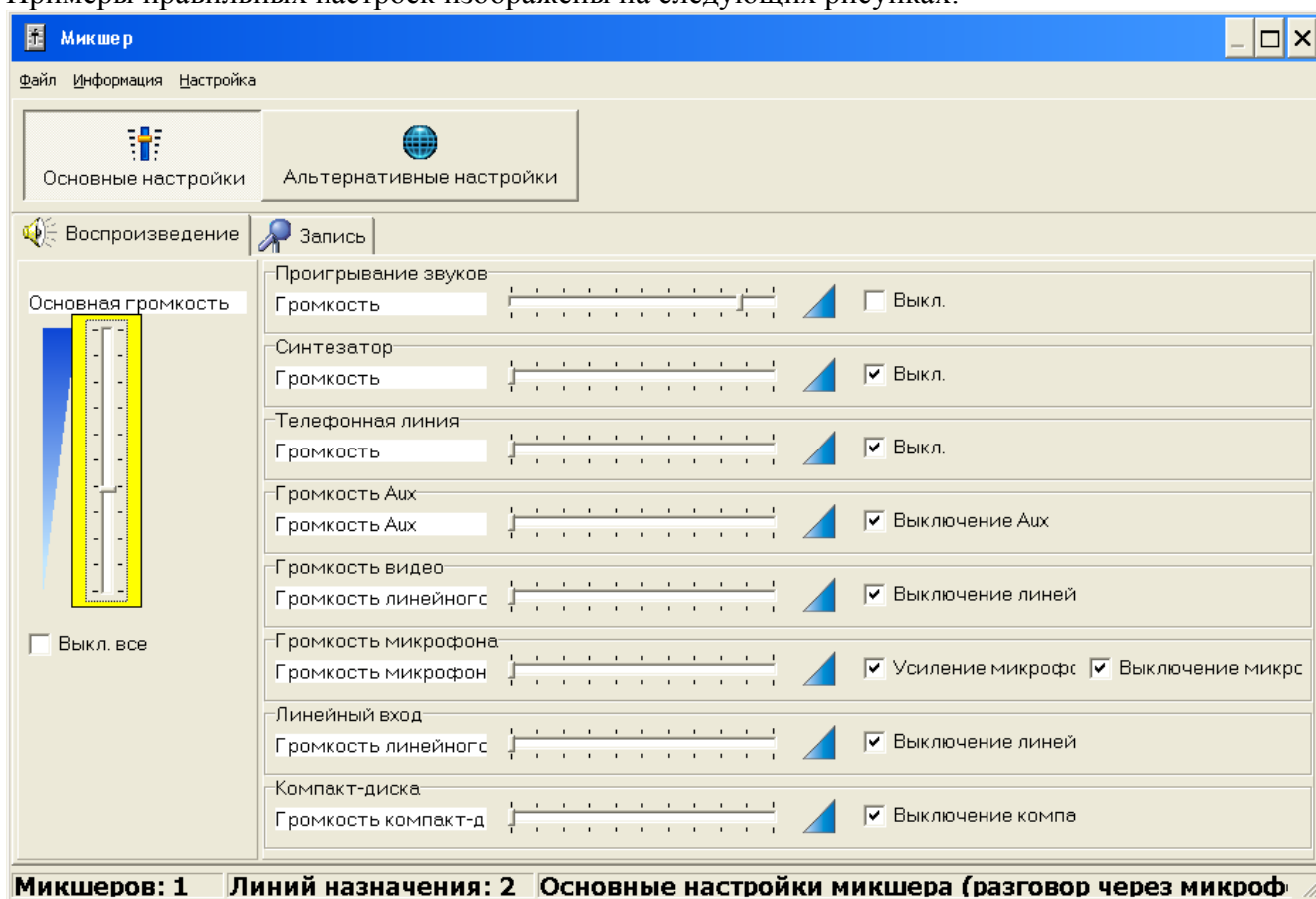
Настройка основного профиля микшера:

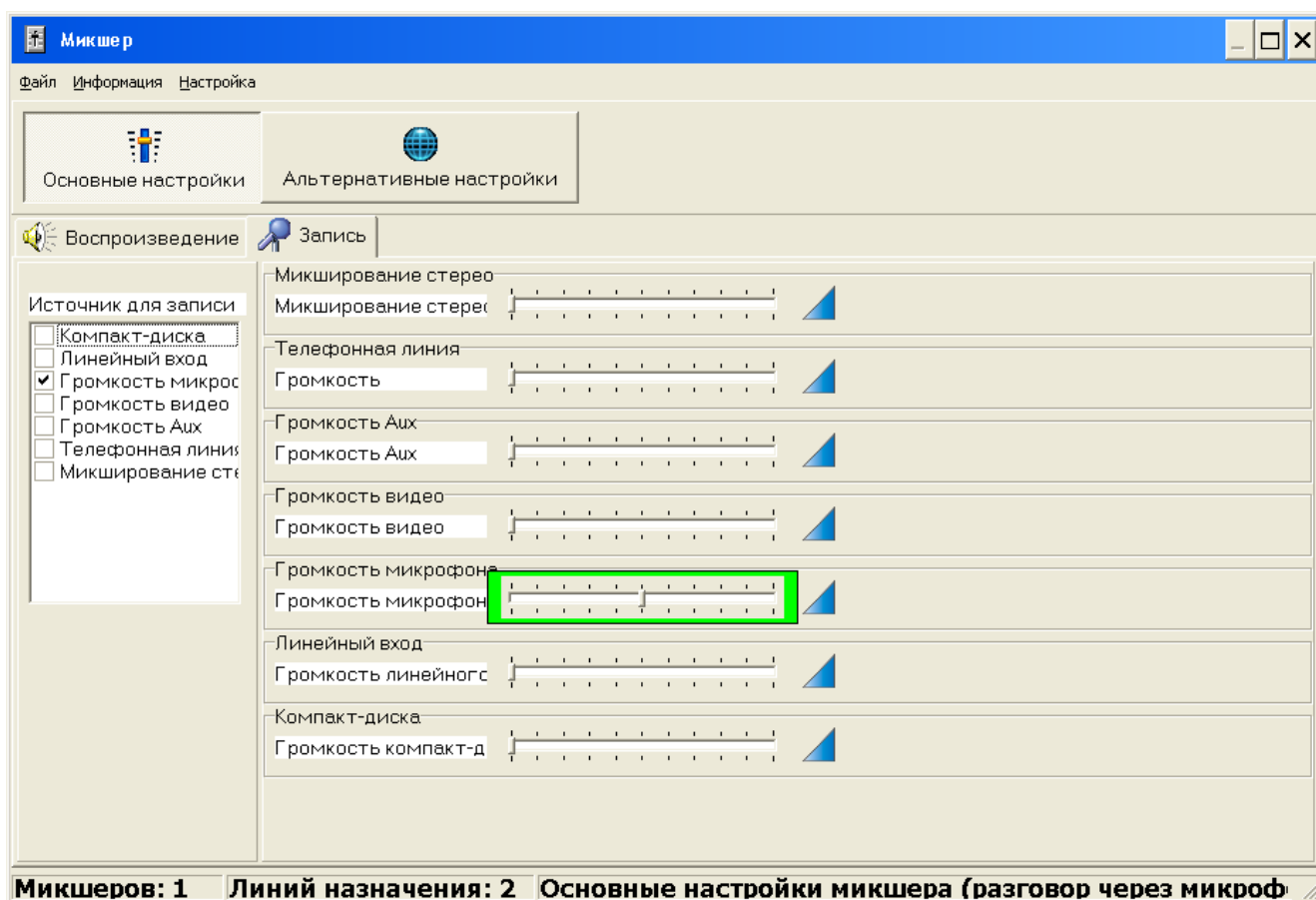
1. Выберите основной профиль настроек. Для этого нажмите кнопку «Основные настройки» в верхней части окна микшера.
2. На вкладке микшера «Запись» («*Recording control*») разрешить запись от источника «Микрофон» («*Microphone*») и установить в среднее положение регулятор записи «Громкость микрофона» («*Microphone volume*»). В дальнейшем можете уточнить положение этого регулятора во время разговора (в окне разговора есть свой регулятор).
3. Подведите курсор мыши на регулятор записи «Микрофон» («*Microphone*») и нажмите правую кнопку. В появившемся меню выберите «Использовать как управление записью звука (микрофон)». При этом он будет выделен зеленой рамкой.

4. На вкладке микшера «**Воспроизведение**» укажите регулятор «**Основная громкость**» («**Master Volume**») – правой кнопкой щелкнуть по нему и выбрать в появившемся меню «**Использовать как управление воспроизведением звука (основная громкость)**». При этом он будет выделен желтой рамкой. Регулятор установите в среднее положение.
5. Все ненужные регуляторы, кроме «**Основная громкость**» «**Master Volume**» и «**Проигрывание звуков**» «**Wave**» выключите: установите галочку «**Mute**» или «**Выкл.**».
6. Произведенные настройки сохраняться автоматически при закрытии окна микшера.

У микшера есть функция автоматической настройки. В большинстве случаев она позволяет автоматически настроить микшер. Для запуска автоматической настройки микшера выберите пункт меню «Файл / Выполнить автонастройку».

Примеры правильных настроек изображены на следующих рисунках:





Альтернативный профиль используется для специальных задач и активизируется программно вызовом функции `H323LoadMixerProfile(1)`. Пример использования альтернативного профиля – проигрывание звукового файла в лифт.

Для проигрывания звукового файла в лифт выберите альтернативные профиль микшера, на вкладке «**Recording Control**» установите источник звука «**Stereo Mix**».

Функция скрипта `H323LoadMixerProfile()` осуществляет загрузку нужного профиля настроек микшера. Полное описание функций по работе с IP телефонией приведено в документе «**Руководство по программированию**» (script.pdf).

### Запись переговоров

Для включения записи переговоров, в настройках проекта на вкладке «**Голосовая связь / Запись**» установите галочку «**Включить запись переговоров**». Запись переговоров производится в поддиректорию `.Audio\` проекта. Для просмотра и проигрывания ранее записанных переговоров используйте пункт меню голосового окна «**Разговор / Просмотр звукозаписей...**».

Для сохранения дискового пространства рекомендуется производить запись переговоров в формате «**MPEG Layer-3 16 kBit/s, 8000 Hz, Mono**».

### Программная поддержка



Все функции IP телефонии доступны из программы на скрипте. Имена функций IP телефонии начинаются с префиксе `H323`. Все функции работают асинхронно. Например, функция отправки вызова `H323Call()` возвращает управление немедленно, но процедура дозвона и установления соединения может занять несколько секунд. Для удобной работы с функциями IP телефонии в программе можно определить обработчик событий от клиента IP телефонии. Все функции, обработчик событий, коды ошибок подробно описаны в руководстве по программированию.



Проект с примерами использования IP телефонии включен в поставку АРМ LanMon и называется «Примеры работы с IP телефонией».

## Режимы APM LanMon

APM LanMon имеет два режима работы:

- Режим редактирования проекта. Режим по умолчанию. Позволяет редактировать ресурсы, писать программы и осуществлять отладку проекта. Любые изменения в проект могут вноситься только в этом режиме. Для временного перехода в режим выполнения проекта нажмите кнопку  на панели инструментов. Для остановки выполнения проекта нажмите кнопку .
- Режим выполнения проекта. При запуске APM LanMon в этом режиме все панели и меню настроек недоступны. Появляется на экране главная карта проекта. Главная карта – это первая карта в списке карт в настройках проекта. Дальнейшая логика работы проекта задается в программе на этапе редактирования. Для запуска APM LanMon в этом режиме надо запустить ярлык, созданный менеджером проектов.

## Настройки проекта

Настройки проекта задают параметры и режимы работы АРМ LanMon. Для вызова окна настроек выберите пункт главного меню «Файл/Настройки проекта...» - появится окно с несколькими вкладками. Вкладка «Сервер» содержит параметры, отвечающие за взаимодействие с сервером LanMon:

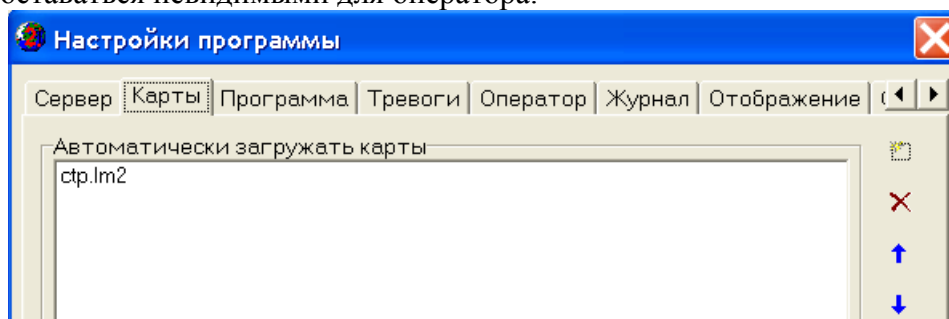
Вкладка «Сервер» определяет взаимодействие с сервером LanMon:





- «Режим работы с сервером»
  - «Не использовать» – сервер LanMon не используется. Все остальные параметры на данной вкладке тоже не используются.
  - «Клиент» - получать состояния каналов только от сервера LanMon. Для сервера мы выступаем как клиент.
  - «Опрос» - получать состояние каналов только от драйверов оборудования и отправлять на сервер LanMon. Для сервера мы выступаем как опросчик.
  - «Клиент + опрос» - получать состояния каналов от драйверов оборудования и от сервера LanMon. Каналы от драйверов оборудования отсылаем на сервер. Для сервера мы выступаем как опросчик и как клиент одновременно.
- «Главный сервер». Если сервер один – то он всегда главный.
  - «Адрес» - имя узла или IP адрес
  - «Порт» - порт сервера LanMon (по умолчанию - 3000)
- «Резервный сервер». Работа в конфигурации с двумя серверами такова: после старта АРМ LanMon подключается к главному серверу и работает с ним. Если подключиться к главному серверу не удалось или связь с ним прервана – производится подключение к резервному серверу. При работе с резервным сервером постоянно проверяется возможность связи с

главным сервером и, как только она появляется, производится отключение от резервного и подключение к главному серверу.

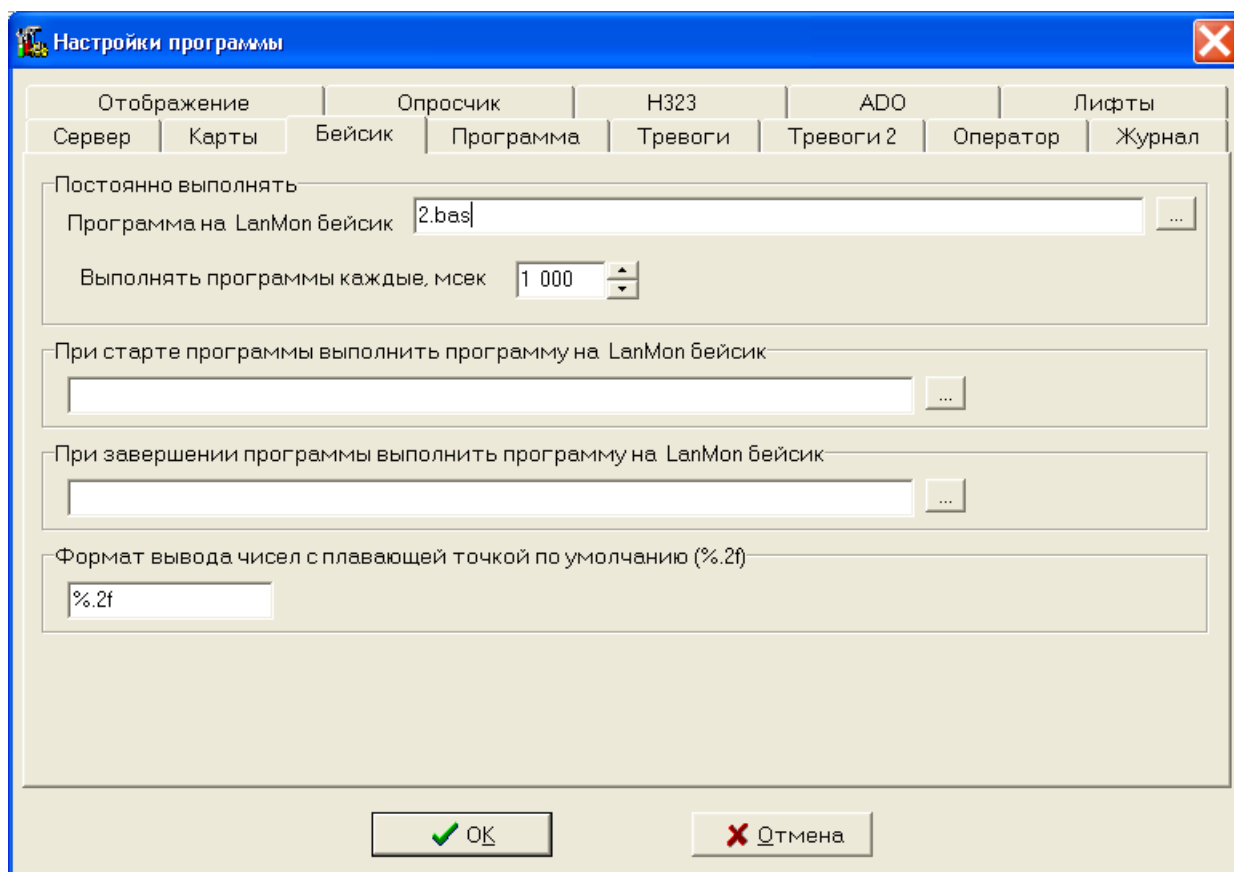
- «Адрес» - имя узла или IP адрес
- «Порт» - порт сервера LanMon (по умолчанию - 3000)
- «Регистрация на сервере». Если сервер один – то он всегда главный.
  - «Имя» - имя пользователя на сервере LanMon
  - «Пароль» - пароль на сервере LanMon
  - «Получать каналы с сервера» - при работе с сервером LanMon есть возможность делать запрос дерева каналов с сервера или использовать локальное дерево каналов. Основное использование данной галочки: на медленных каналах связи данную галочку отключают для уменьшения трафика при старте APM LanMon.
  - «Синхронизировать время с сервером» - брать время с сервера LanMon и устанавливать его на компьютере. Если сервер LanMon выполняется на том же компьютере, что и APM LanMon – эта галочка должна быть снята.
- «Таймаут команд».
  - «Обычная команда» - таймаут при обмене с сервером по TCP/IP. Значение по умолчанию – 10 сек.
  - «Проверка связи с сервером» - Таймаут для проверки связи с сервером в случае простоя. Если такая проверка не удалась – APM LanMon отсоединяется от сервера. Если задан резервный сервер – пытаемся соединиться с ним. Если не задан резервный сервер – пытаемся снова подсоединиться к главному серверу до тех пор, пока это не удастся. Значение по умолчанию – 300 сек.

Вкладка «Карты» содержит список карт, входящих в текущий проект (т.е. эти карты станут доступны после загрузки APM LanMon). Причем первая карты в списке – это главная карта проекта. Она появляется на экране после загрузки APM LanMon автоматически. Все остальные карты могут оставаться невидимыми для оператора.



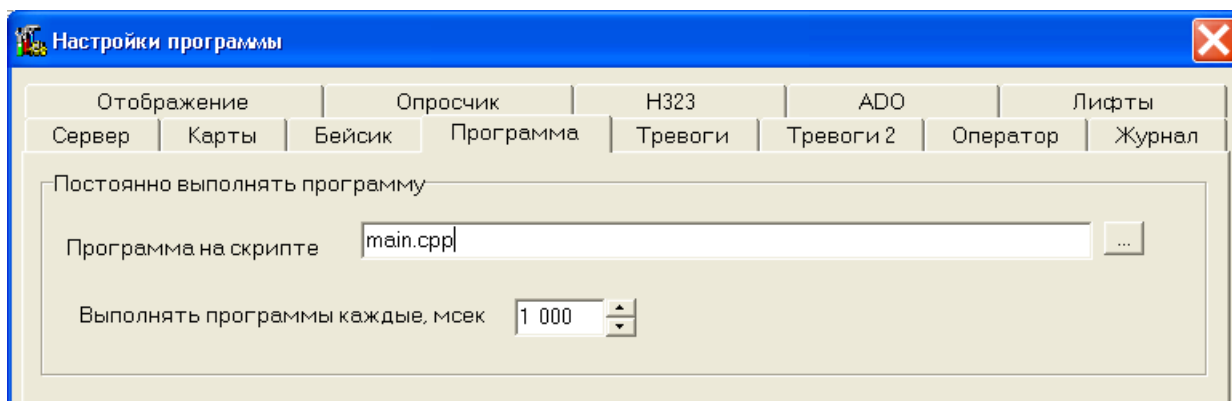
- Кнопка  добавляет карту в список карт. Если полный путь не задан, то подразумевается стандартная поддиректория проекта для карт: ..\MAP\ . Настоятельно рекомендуется хранить карты проекта в стандартной директории.
- Кнопка  удаляет карту из списка.
- Кнопка  перемещает карте вверх по списку. Карты загружаются в порядке следования в списке.
- Кнопка  перемещает карте вниз по списку.

Вкладка «Бейсик» содержит настройки, относящиеся к работе программ на бейсике:



- «При старте программы выполнить» - задается имя файла с программой на бейсике, которая выполнится сразу после старта программы. Если полный путь файла с программой не задан - подразумевается директория для программ на бейсике по умолчанию: `..\BASIC\`. Настоятельно рекомендуется хранить все программы проекта в директории по умолчанию.
- «Постоянно выполнять» - задается имя файла с программой, которая выполняется постоянно с указанным интервалом.
- «При завершении программы выполнить» - задается имя файла с программой, которая выполняется непосредственно перед завершением работы APM LanMon.
- «Формат вывода чисел с плавающей точкой по умолчанию (%.2f)» - по умолчанию программы на бейсике осуществляют вывод чисел с плавающей точкой в этом виде. Формат задается по правилам языка C. «%.2f» означает, что мы хотим 2 цифры после точки.

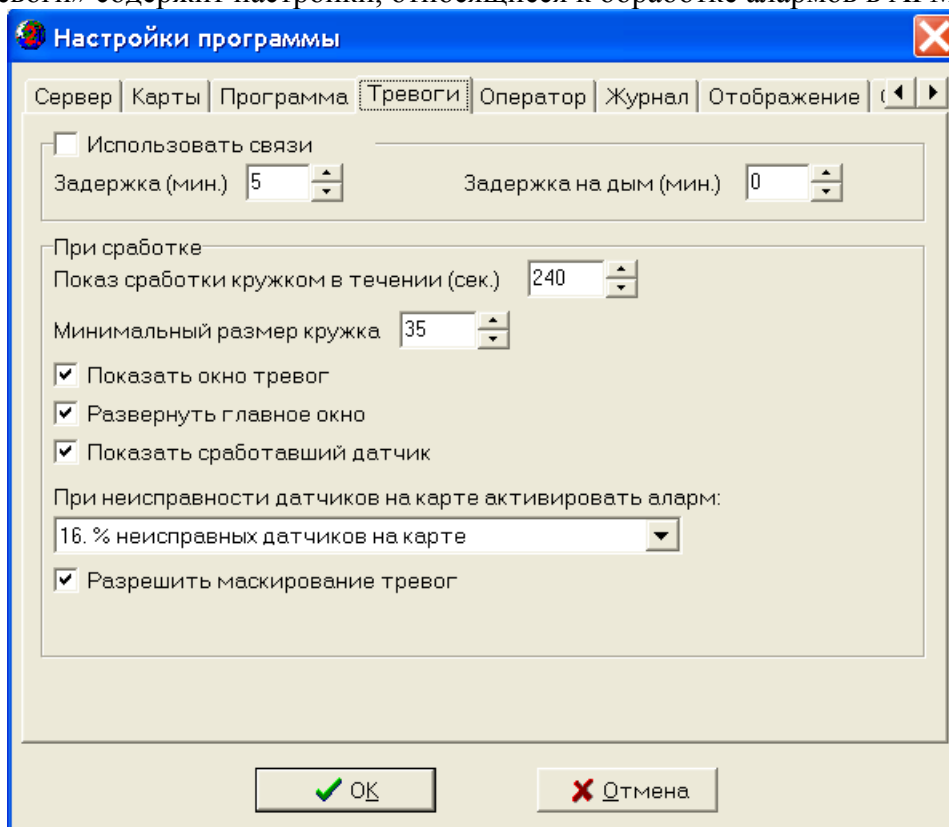
На вкладке «Программа» задаются настройки программы на скрипте:



«Программа на скрипте» - главная программа проекта. Она будет содержать все пользовательские функции и обработчики событий. Язык программирования на котором написана программа различается по расширению. Например: `spr` – язык C++, `pas` – паскаль и т.д.

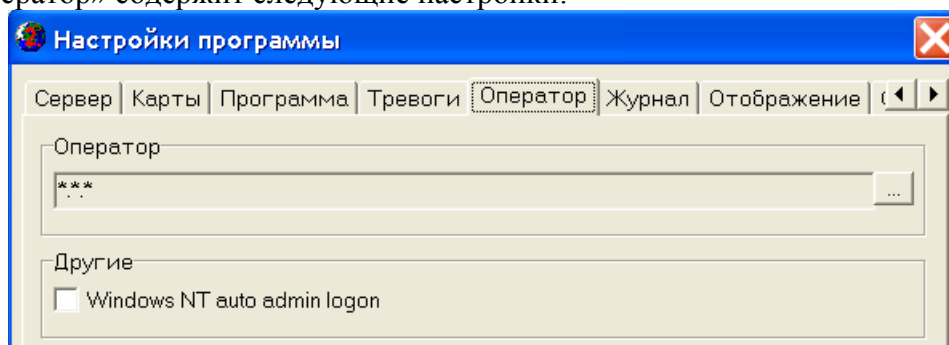
«Выполнять основную процедуру каждые, мсек» - периодичность выполнения основной процедуры скрипта.

Вкладка «Тревоги» содержит настройки, относящиеся к обработке алармов в АРМ LanMon:



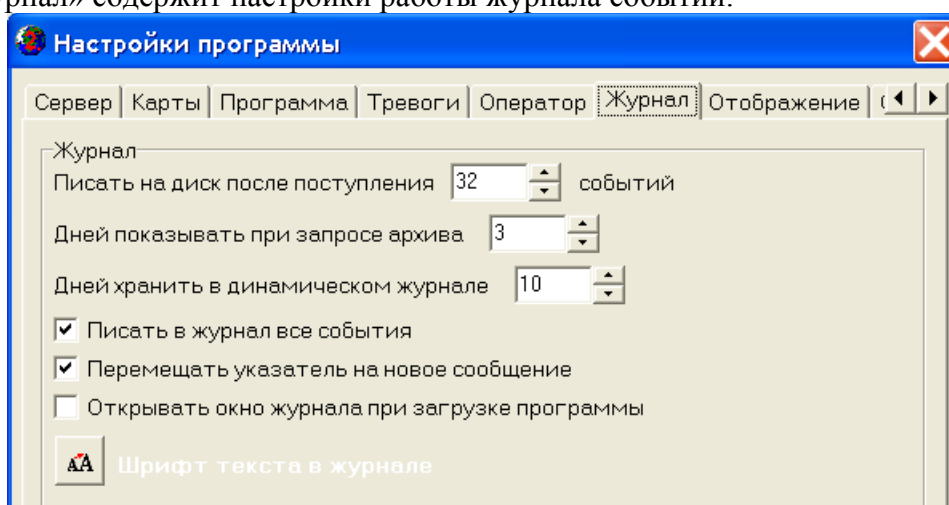
- «Использовать связи» - Задействовать механизм связей при активизации дискретных алармов.
- «Показ срабатывания кружком в течении» - при активизации аларма, отображаемый объект на карте обводится красным кружком. Кружок уменьшается и исчезает совсем в течение этого времени.
- «Минимальный размер кружка» - задает минимальный диаметр кружка в точках. Удобно использовать, когда на карте есть маленькие объекты (5 на 5 точек или меньше).
- «Показать окно тревог» - при активизации аларма вывод окна тревог.
- «Развернуть главное окно» - при активизации аларма развернуть главное окно АРМ LanMon если оно была свернута.
- «Показать сработавший датчик» - при активизации аларма показать карту, на которой находится данный объект и переместить объект в область видимости окна при наличии полос прокрутки.
- «При неисправности датчиков на карте активизировать аларм» - если число неисправных датчиков на карте превысило предел, установленный в параметрах карты – активизируем указанный аларм. Аларм должен быть заранее настроен в редакторе алармов.
- «Разрешить маскирование тревог» - у каждого отображаемого объекта на карте в контекстное меню (вызываемое по щелчку правой клавишей мыши) включить пункт «маскирование».

Вкладка «Оператор» содержит следующие настройки:



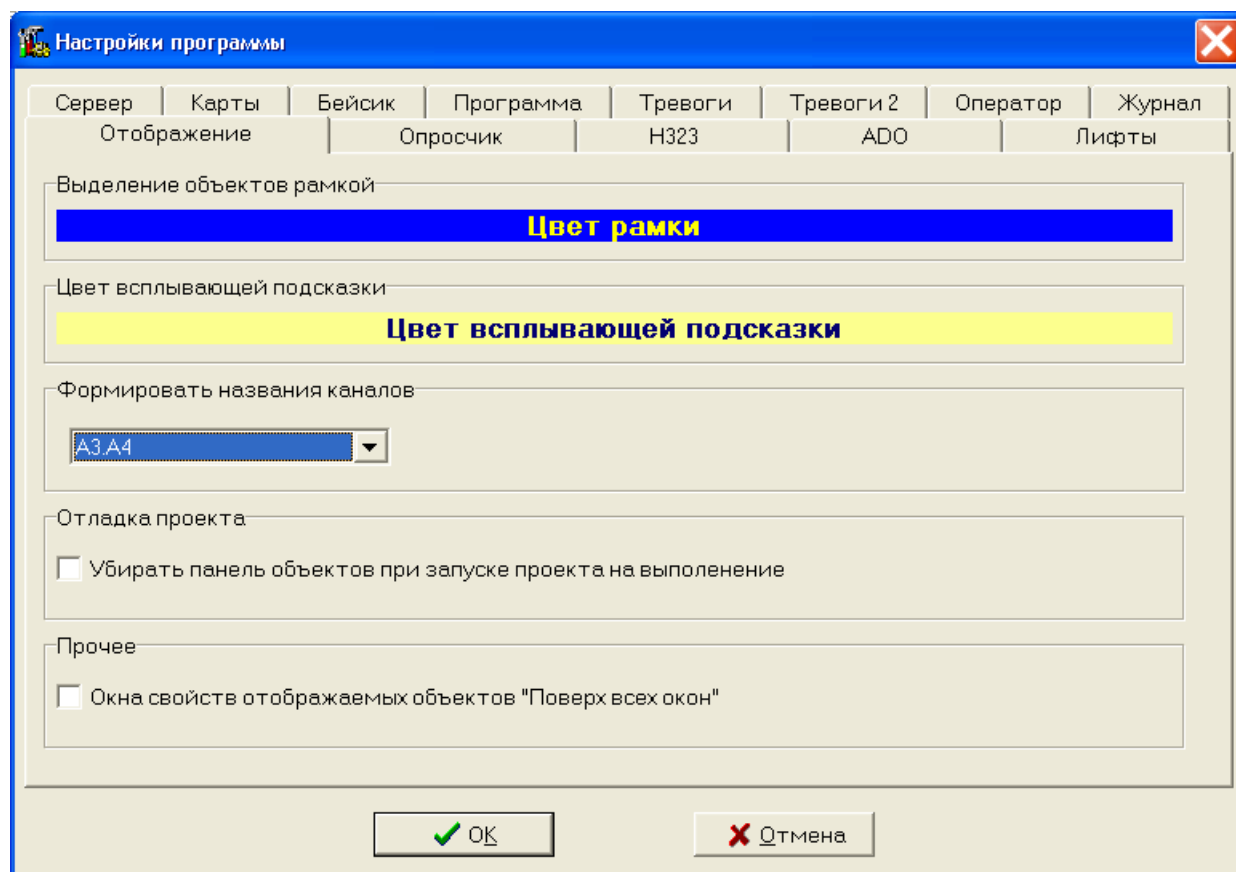
- «Оператор» - канал типа «оператор». От имени этого канала производится регистрация в журнале событий всех системных событий.
- «Windows NT auto admin logon» - автологон в Windows NT 4.0.

Вкладка «Журнал» содержит настройки работы журнала событий:



- «Писать на диск после поступления N событий» - после поступление N событий происходит их принудительная запись на диск (т.е. при сбое питания они не исчезнут). Чем меньше это число, тем надежнее хранение журнала и медленнее работа АРМ LanMon. Разумные значения этого параметра зависят от интенсивности потока событий и лежат в диапазоне от 10 до 200.
- «Дней показывать при запросе архива» - при операциях просмотра выборки значений по одному каналу (с помощью пункта контекстного меню отображаемого объекта «Сообщения...») показывать данные из архива за это количество дней.
- «Дней хранить в динамическом журнале» - максимальная глубина сообщений в журнале событий.
- «Писать в журнал все события» - надо писать в журнал все события. По умолчанию эта галочка установлена.
- «Перемещать указатель на новое сообщение» - если эта галочка установлена, то при приходе нового сообщения курсор в журнале переходит на последнее сообщение.
- «Открывать окно журнала при загрузке программы» - если эта галочка установлена, то при старте АРМ LanMon окно журнала автоматически появляется на экране.
- Кнопка позволяет настроить шрифт текста в журнале событий.

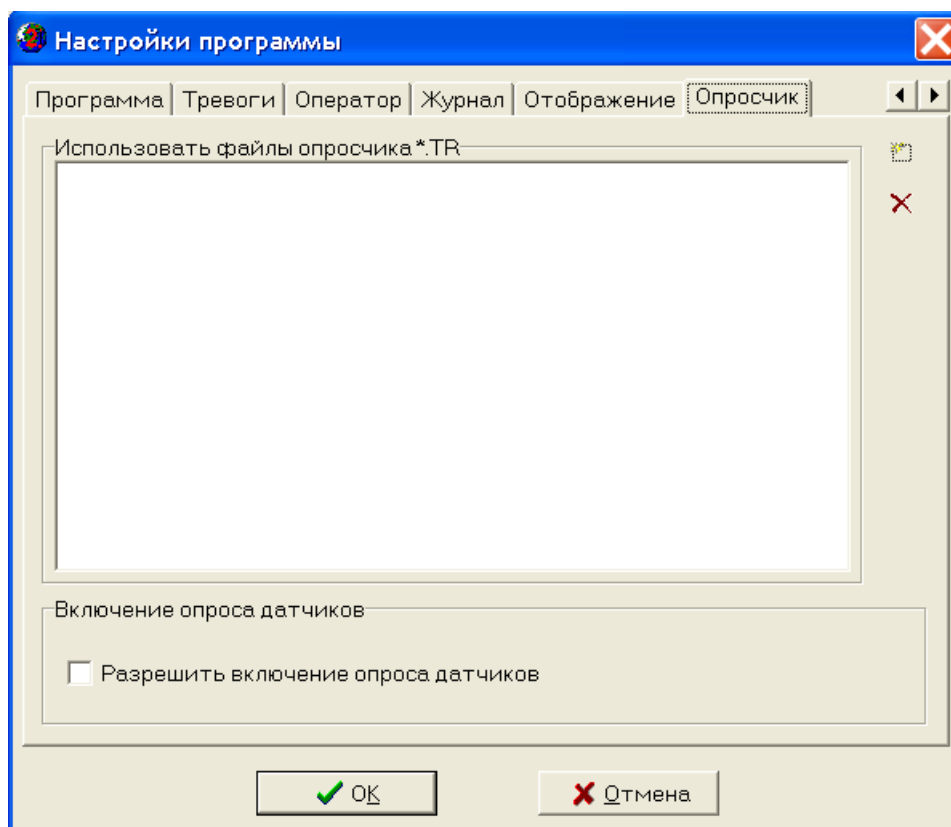
Вкладка «Отображение» содержит следующие настройки:





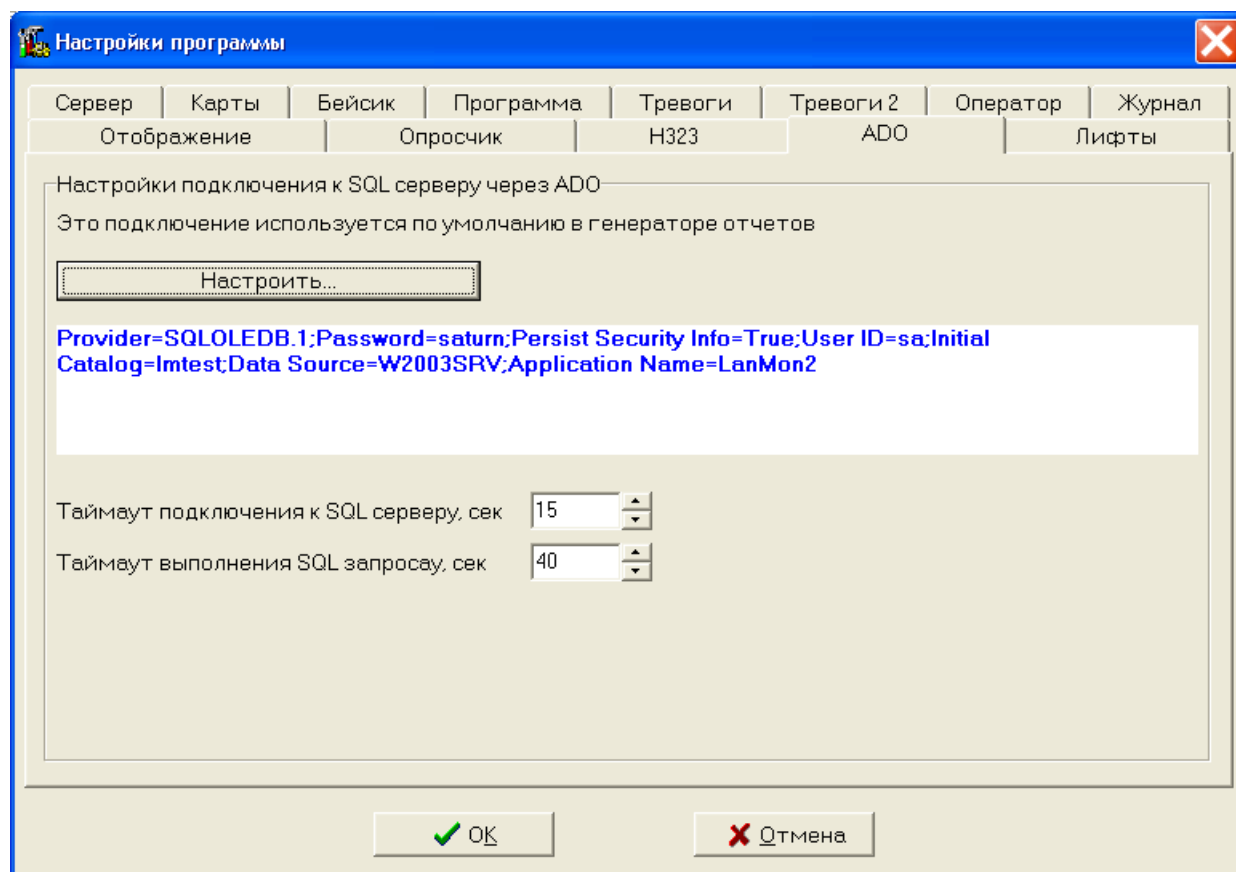
- «Выделение объектов рамкой» - назначается цвет рамки выделения объектов в режиме выполнения проекта.
- «Цвет всплывающей подсказки» - понимается однозначно.
- «Формировать название каналов» - маска для формирования названий каналов по умолчанию.

Вкладка «Опросчик» содержит настройки для получения информации из файлов настроек опросчиков СОС-95 \*.tr:





- Кнопка  добавляет файл опросчика СОС-95 \*.tr в список используемых в АРМ LanMon. Если хоть 1 файл добавлен в список в контекстном меню отображаемого объекта - на карте появляется пункт «Физические параметры...». При выборе этого пункта появляется окно с параметрами физического подключения датчика к контроллеру (адрес и т.п.).
- Кнопка  удаляет файл опросчика СОС-95 \*.tr из списка используемых в АРМ LanMon.
- «Разрешить включение опроса датчиков» - если поставить эту галочку, то у отображаемого объекта появляется пункт контекстного меню «Включить опрос». Эту команду обрабатывает опросчик СОС-95 и включает опрос датчика.




На вкладке «ADO» задаются параметры подключения к источнику данных ADO (например, к SQL серверу). Это подключение может быть использовано в отчетах и в программе на скрипте для доступа к данным.

На вкладке «Лифты» задается имя файла с настройками лифтовых сигналов. Этот файл должен быть сгенерирован при наладке лифтовой системы программой RASOS. Наличие этого файла дает правильную расшифровку имен сигналов у каналов типа «Лифт».

## Журнал событий

Все каналы, которые APM LanMon получает от сервера или от драйверов оборудования регистрируются в журнале APM LanMon. Для открытия журнала выберите пункт меню «Просмотр/Журнал...». Появится окно следующего вида:



Дата	Время	Объект	Расположение	Оборудование	Состояние
24.03	18:04:10	ВКК	Сервис	Качество в луче СОС-95	Состояние не определено
24.03	18:04:10	ВКК	Сервис	ВКК корп.106	Состояние не определено
24.03	18:07:35	ВКК	ПК0	A/B	Закрыта
24.03	18:07:49	ВКК	ПК0	A/B	Открыта
24.03	18:07:58	ВКК	ПК0	A/B	Закрыта
24.03	18:08:05	ВКК	ПК0	A/B	Открыта

☒ Переход на новое сообщение

Координаты окна журнала на экране запоминаются автоматически. Каждая строка журнала это изменение состояния одного канала (одно сообщение). Строки выделяются разным цветом, в зависимости от статуса сообщения. Цвета распределяются так:

- Красный – тревожные сообщения (данное сообщение активизировало один из установленных алармов);
- Желтый – сообщение о срабатывании датчика без тревоги (например: сработал охранный извещатель когда объект был снят с охраны);
- Зеленый – служебные сообщения;
- Серый – сообщения о неопределенном состоянии канала или о неисправности;
- Белый - все остальные сообщения;

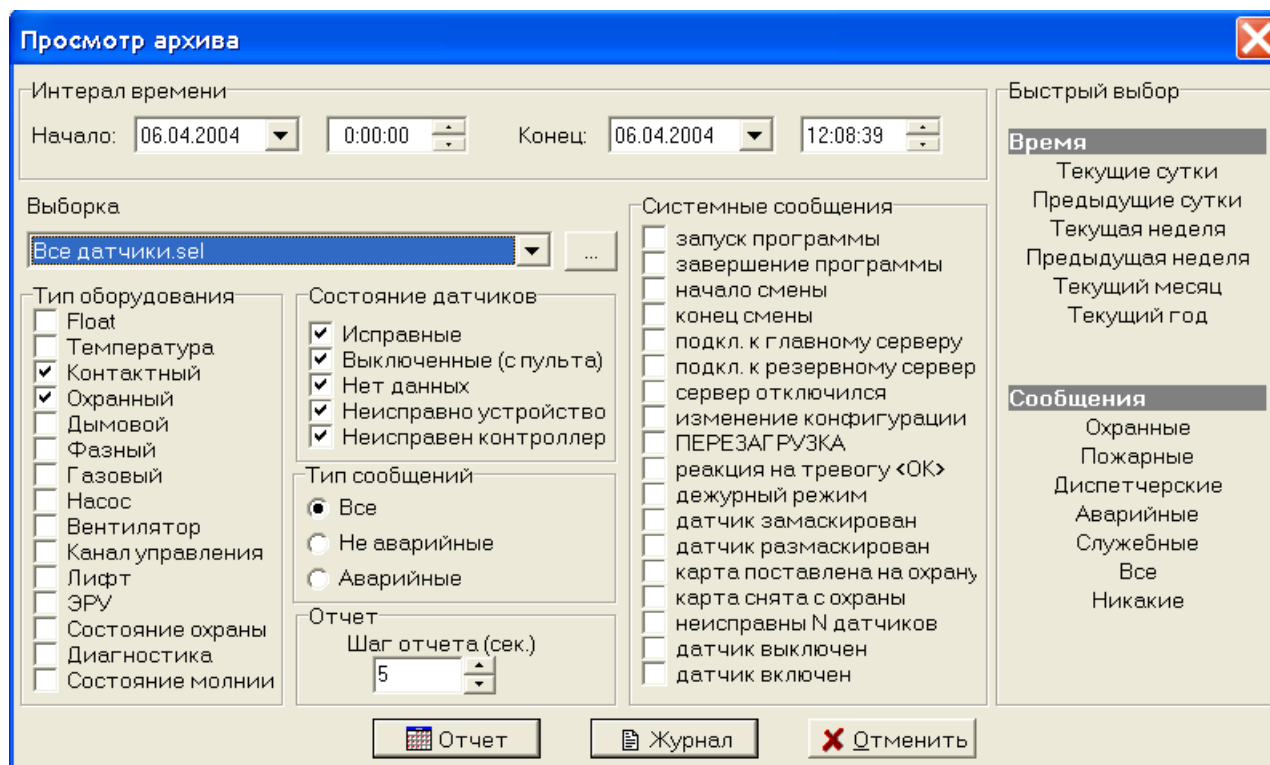
Если установлена галочка «переход на новое сообщение», то при регистрации нового сообщения курсор в журнале перескочит вниз - на него. Если нужно просмотреть журнал за более ранний период, снимите эту галочку, чтобы появление новых сообщений не мешало просмотру.

В системном меню журнала есть пункт «Шрифт...». Он позволяет настроить шрифт текста в журнале. В настройках APM LanMon есть вкладка «Журнал», которая содержит некоторые настройки журнала.

Окно журнала можно скрыть, нажав крестик в правом верхнем углу. Функция бейсика «*LMViewLog()*» показывает окно журнала на экране.

## Просмотр архива

Все сообщения, поступающие от сервера или от драйвера оборудования, регистрируются в архиве APM LanMon. Архив расположен в поддиректории проекта `..\LOG\` и представляет собой файлы в формате `csv`. Все колонки в файлах архива формализованы. За каждые сутки ведется отдельный файл. Название файла формируется так: «`дд_мм_гг.csv`». Сообщения пишутся в архив без ограничений, файлы архива не удаляются. Для просмотра архива выберите пункт меню «Просмотр/Архив...». Аналогично: функция скрипта «`LMViewArchive()`» показывает окно просмотра архива на экране.



Для просмотра архива необходима «выборка каналов». Выборка представляет собой список каналов или масок, информацию по которым необходимо выбрать из архива.

Архив сообщений можно просмотреть в двух видах:

- В виде журнала событий
- В виде отчета в формате Microsoft Excel

Для просмотра архива в виде журнала сообщений:

- задайте интервал времени;
- выберите выборку каналов;
- выберите нужные типы оборудования;
- выберите нужные состояния датчиков;
- выберите нужный тип сообщений;
- выберите нужные системные сообщения;
- нажмите кнопку «Журнал».

Откроется новое окно по виду идентичное окну журнала событий.

Отчет по срабатываниям предназначен для представления срабатываний охранных, пожарных, газовых и прочих извещателей в виде графика. Для формирования данного отчета на компьютере должен быть установлен Microsoft Excel 2002 или старше. Для просмотра данных из архива в виде отчета по срабатываниям датчиков:

- задайте интервал времени;
- выберите выборку каналов;
- выберите нужные типы оборудования;
- выберите все состояния датчиков;
- выберите тип сообщений «Все»;
- нажмите кнопку «Отчет».

При этом загрузится Microsoft Excel с документом следующего вида:

Дата Время	ОДБ Зона 5	ОДБ Зона 4	ОДБ Зона 3	ОДБ Зона 2	ОДБ Зона 1	ОД1 Зона 1	ОД1 Зона 2	ОД1 Зона 3	ОД1 Зона 4	ОД1 Зона 5	ОД1 Зона 6	ОД1 Зона 7	ОД1 Зона 8	ОД1 Зона 9	ОД1 Зона 10	ОД1 Зона 11	ОД1 Зона 12	ОД1 Зона 13	ОД1 Зона 14	ОД1 Зона 15	ОД2 Зона 1	ОД2 Зона 2	ОД2 Зона 3	ОД2 Зона 4	ОД2 Зона 5	ОД2 Зона 6	ОД2 Зона 7	ОД2 Зона 8	ОД2 Зона 9	ОД2 Зона 10	ОД2 Зона 11	Всего
15.01.2004 0:07:38																																2
15.01.2004 18:49:13																																0
15.01.2004 18:49:14																																3
15.01.2004 18:49:18																																0
15.01.2004 18:49:19																																2
15.01.2004 18:49:21																																0
15.01.2004 18:49:23																																2
16.01.2004 12:07:07																																0
16.01.2004 12:07:08																																3
16.01.2004 12:07:10																																0
16.01.2004 12:16:54																																2
16.01.2004 12:16:55																																0
16.01.2004 12:16:56																																2
16.01.2004 12:16:58																																0
16.01.2004 12:16:59																																2
16.01.2004 12:17:01																																0
16.01.2004 12:17:03																																2
16.01.2004 12:17:04																																0
16.01.2004 12:17:06																																2
16.01.2004 12:17:11																																0
16.01.2004 12:17:14																																1
16.01.2004 12:17:16																																2
16.01.2004 12:17:20																																0
16.01.2004 12:17:21																																0
16.01.2004 12:17:24																																1
16.01.2004 12:17:44																																0
16.01.2004 12:17:46																																2
16.01.2004 12:17:48																																1
16.01.2004 12:17:50																																0
16.01.2004 12:17:52																																1
16.01.2004 12:17:57																																2
16.01.2004 12:17:59																																0
16.01.2004 12:18:01																																1
16.01.2004 12:18:03																																0
16.01.2004 12:18:05																																3
16.01.2004 12:18:07																																4
16.01.2004 12:18:09																																2
16.01.2004 12:18:11																																4
Всего	0	0	0	0	0	3	2	4	2	1	1	2	1	1	2	1	1	3	4	1	2	3	1	2	1	2	1	2	1	1	1	46

В заголовке колонки – название канала. Количество колонок соответствует содержанию выборки каналов. Ячейки с цифрами – это количество срабатываний канала. Если ячейка залита желтым цветом – это срабатывание, когда система не стояла на охране (для охранных датчиков). Если ячейка залита красным цветом – это тревожное извещение. Заголовки колонок имеют цвета:

- Желтый – в данной колонке нет ни одного срабатывания;
- Зеленый – в данной колонке всего одно срабатывание;
- Белый – в данной колонке более 2-х срабатываний;

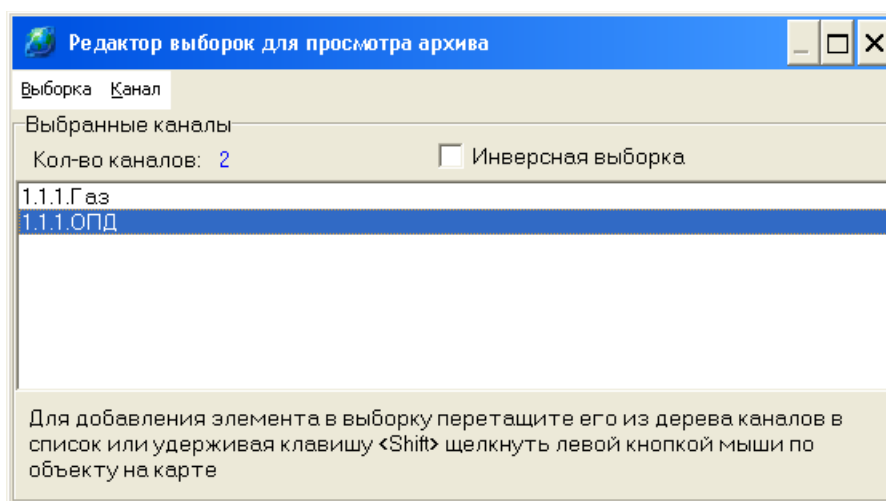
Для формирования отчета по срабатываниям нужна выборка, состоящая из каналов (маски в выборке недопустимы). Это объясняется необходимостью установить порядок следования каналов.

Для создания выборки каналов воспользуйтесь редактором выборок. Для вызова редактора выборок откройте пункт меню «Ресурсы/Редактор выборок...».

## Редактор выборок

Выборкой называется именованный список адресов каналов (или масок каналов). Маска каналов представляет собой адрес, в котором хоть один элемент равен нулю. Нулевой адрес соответствует любому числу. Т.е. если в выборке присутствует маска 1.1.1.0, то ей соответствуют каналы с адресами 1.1.1.1, 1.1.1.2, 1.1.1.3 и т.д.

Выборка хранится в файле с расширением «.sel» в поддиректории «TEMPLATE» текущего проекта. Редактор выборок позволяет создавать список каналов для последующего запроса данных из архива. Для вызова редактора выборок откройте пункт меню «Ресурсы/Редактор выборок...». Вызов редактора выборок доступен только в режиме редактирования проекта.



Для добавления канала в выборку, перетащите его из дерева каналов в список или, удерживая клавишу «Shift», щелкните левой клавишей мыши по объекту карты. Один и тот же канал можно добавить в выборку только один раз. При перетаскивании из дерева каналов маски (адреса хоть с одним нулем) предлагается добавить саму маску или список каналов по этой маске.

Выборка может содержать и маски каналов. Звездочка в названии канала означает любой адрес. Для создания выборки «все каналы» выберите пункт меню «Канал \ Добавить все каналы».

Для удаления канала из выборки выделите канал в списке и нажмите клавишу «Delete» или в меню выберите «Канал \ Удалить».

Для перемещения выбранного канала в списке выше или ниже используйте комбинации клавиш «Shift+стрелка вверх» или «Shift+стрелка вниз».

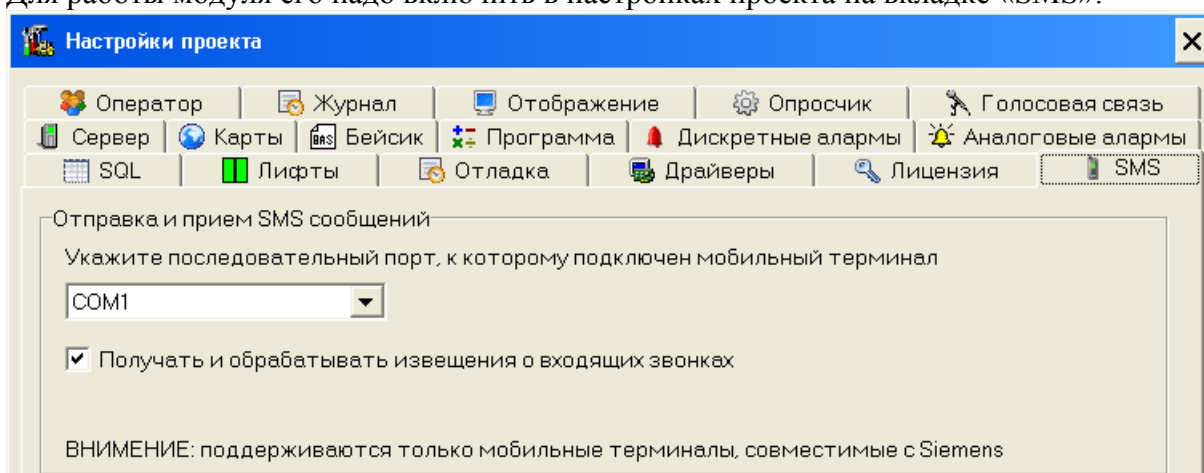
Для сохранения выборки в файле выберите в меню «Выборка \ Сохранить...». Выборки надо сохранять в файлах с расширением «.sel» в поддиректории «TEMPLATE» текущего проекта. Если сохранить выборку в другое место, то при просмотре архива она окажется недоступной.

Если установить галочку «Инверсная выборка», то выборка будет включать все каналы, кроме перечисленных в списке.

## Модуль GSM

Модуль GSM предназначен для приема и отправки SMS сообщений и регистрации входящих голосовых вызовов. Данный модуль всегда включен в комплект поставки АРМ LanMon, но для его работы требуется отдельная лицензия, которую надо приобрести. Модуль работает только с Siemens совместимым мобильным телефоном или мобильным терминалом, подключенным к компьютеру по последовательному интерфейсу. Все функции модуля GSM доступны из программы на скрипте.

Для работы модуля его надо включить в настройках проекта на вкладке «SMS»:



Надо указать последовательный порт компьютера, к которому подключен мобильный терминал. Для регистрации входящих голосовых вызовов установите галочку «Получать и обрабатывать извещения о входящих звонках».

Рекомендуемый мобильный терминал для работы с модулем – Siemens ES75. Рекомендуется производить подключение к компьютеру по интерфейсу RS-232.

Программный интерфейс модуля подробно описан в руководстве по программированию. В комплект поставки АРМ LanMon входит проект «Пример проекта SMS», который демонстрирует работу с модулем.

## Требования к аппаратному и программному обеспечению

Программа APM LanMon работает под управлением следующих операционных систем Windows 98/ME/NT 4.0 SP6/XP/2000 server/2003 server (подразумеваются только русскоязычные версии перечисленных операционных систем). В таблице приведены минимальная и рекомендуемая конфигурации компьютера. Вы должны иметь в виду, что для стабильной и эффективной работы требуется использовать рекомендуемую конфигурацию.

	Минимальная	Рекомендуемая
Процессор	Intel Celeron 2000 MHz	Intel Pentium 4 3000 MHz
Оперативная память	128 Mb	1024 Mb
Разрешение экрана	1024*768 - 24 битный цвет	1280*1024 - 24 битный цвет
Жесткий диск	10 Гб свободно	40 Гб свободно
Свободный USB порт	Да	Да
CD-ROM drive	Опционально	Да
Звуковая карта, колонки	Опционально	Да
Сетевая плата	Опционально	Опционально
Интерфейс RS-232	Опционально	Да

### Примечания:

- Минимальные требования даны для системы с 500 каналами и одним драйвером оборудования. При увеличении количества каналов требования увеличиваются. При применении более одного драйвера оборудования требования увеличиваются.
- Совместное использование APM LanMon параллельно с другими программами также может снизить производительность.

Настоятельно рекомендуется использовать Windows платформы NT: NT 4.0 SP6/XP/2000/2003 и файловую систему NTFS. Функция «Просмотр архива в виде отчета» требует установленный Microsoft Excel версии 2002 или выше.

Работа APM LanMon в иных операционных системах или под управлением эмуляторов Windows не проверялась, и нет никаких гарантий такой работы. Пользователь может запускать APM LanMon в таких средах на свой страх и риск.