



Контроллер  
со встроенным WEB интерфейсом  
программирования и отображения объектов

## **EVIKA Logic Machine 2**

### **Руководство пользователя**



1.1.01.003 2013.03.21

<b>LM2-CON</b>	<b>LogicMachine2 Control FULL</b> Свободно программируемый контроллер
<b>LM2-INT</b>	<b>LogicMachine2 Interface FULL</b> Свободно программируемый контроллер с 3-мя портами RS232
<b>LM2-Reactor</b>	<b>LogicMachine2 Re:actor</b> Свободно программируемый контроллер с увеличенным количеством портов

# СОДЕРЖАНИЕ

Авторские права .....	5
Уведомление .....	5
Товарные знаки .....	5
Вводная часть .....	5
Техническая поддержка .....	5
<b>ТЕРМИНОЛОГИЯ.....</b>	<b>7</b>
<b>ОПИСАНИЕ УСТРОЙСТВА.....</b>	<b>9</b>
Обзор функций и возможностей .....	9
Аппаратное обеспечение.....	9
Программное обеспечение и скрипты .....	9
Технические характеристики.....	11
Семейство продуктов .....	11
Спецификация устройств .....	11
Поддержка стандартов .....	12
Внешний вид контроллеров .....	13
LM2-CON и LM2-INT .....	13
LM2-Re:actor .....	13
Кнопка Reset \ FactoryReset.....	13
Индикаторы LED1 и LED2.....	14
Комплект поставки .....	14
Заводские значения параметров .....	14
<b>УСТАНОВКА .....</b>	<b>16</b>
Включение и отключение питания .....	16
<b>ПРОГРАММИРОВАНИЕ И НАСТРОЙКА .....</b>	<b>17</b>
Совместимость с браузерами .....	17
Подключение к WEB интерфейсу .....	17
Вы подключены к сети устройства.....	17
Устройство подключено к другой сети .....	17
Селектор Интерфейсов .....	18
Сервер данных. ....	19
Права доступа к интерфейсам и функциям .....	19
Типы Пользователей.....	19
<b>NETWORK CONFIGURATION (FLASHSYS) .....</b>	<b>20</b>
Назначение FlashSYS.....	20
Начальная страница FlashSys .....	20
Меню FlashSys .....	20
Релиз FlashSys .....	20
Load Average статус (Загрузка центрального процессора) .....	20
FlashSys\System (Системные настройки и программы) .....	21
FlashSys\System\Packages (Библиотеки программ и данных) .....	21
FlashSys\System\GUI login (Назначение паролей доступа) .....	22
FlashSys\System\Backup and restore (Резервные копии параметров FlashSys) .....	22
FlashSys\System\Upgrade firmware (Загрузка заводского ПО) .....	23
Определение текущих релизов установленного заводского ПО .....	24
FlashSys\System\Reboot (Перезагрузка). ....	24
FlashSys\System\Shutdown (Подготовка к отключению питания).....	24
FlashSys\Networks.....	24
FlashSys\Networks\Interfaces (Доступные сетевые подключения) .....	25

FlashSys\Networks\Interfaces\Ethernet (Ethernet подключения) .....	25
FlashSys\Networks\Interfaces\Wireless (Беспроводные подключения) .....	26
Параметры подключения Ethernet .....	27
FlashSys\Network\Routes (Таблица маршрутизации) .....	28
FlashSys\Network\KNX connection (параметры KNXIP роутера) .....	29
FlashSys\Network\BACnet (подключение к сети BACnet) .....	31
FlashSys\Services (Сетевые сервисы) .....	31
FlashSys\Services\FTP Server (Сервер FTP) .....	31
FlashSys\Services\System Monitoring (Скрипт системного мониторинга) .....	31
FlashSys>Status (Системная статистика и мониторинг) .....	33
FlashSys>Status\System Status (Монитор системных ресурсов) .....	33
FlashSys>Status\Network Status (Статус подключений) .....	34
FlashSys>Status\Network Utilites (Сетевые сервисные программы) .....	34
FlashSys>Status\System Log (Протокол работы ОС) .....	35
FlashSys>Status\Running processes (Монитор процессов) .....	35
FlashSys>Status\Help (Помощь) .....	35
<b>WEBI LOGIC MACHINE (ИНТЕРФЕЙС ИНСТАЛЛЯТОРА) .....</b>	<b>36</b>
Общая концепция программирования Logic Machine 2 .....	36
Технология. ....	36
Создание «поэтажной» структуры и добавления объектов на план .....	37
Создание структуры «здание/этаж» и добавление объектов на план .....	37
Logic Machine 2 для настройки конфигурации сети KNX/EIB через протокол TCP/IP .....	37
Быстрый запуск. Создание визуализации для iPad/PC .....	39
Импорт объектов .....	39
Создание «поэтажной» структуры и добавления объектов на план .....	40
Создание структуры «здание/этаж» и добавление объектов на план .....	40
Добавьте объекты на созданный план этажа .....	42
Настройка визуализации на PC, Mac .....	43
Вход в графический интерфейс пользователя .....	45
Окно Logic Machine 2 .....	46
Меню Scripting .....	46
Добавление нового скрипта .....	47
Сценарии, основанные на событиях .....	47
Резидентные скрипты .....	49
Запланированные скрипты .....	50
Инструменты .....	51
Редактор скриптов. Синтаксис .....	53
Функции объектов .....	54
Функции, связанные с объектом .....	55
Функции для работы с групповыми адресами .....	55
Типы данных .....	57
Общие функции. Хранение данных. Предупредительные сообщения .....	57
Хранение данных .....	57
Примеры .....	58
Функция формирования предупредительных сообщений .....	59
Формат запланированного времени выполнения скрипта .....	59
<b>РАСШИРЕННАЯ БИБЛИОТЕКА ФУНКЦИЙ .....</b>	<b>60</b>
Объекты .....	62
Параметры объектов .....	63
Параметры визуализации объекта .....	64
Изменение состояния объекта .....	65
Журнал событий .....	66
Экспорт журнала .....	66
Здания .....	69
Визуализация .....	71
Редактор плана этажа здания .....	72
Объекты системы управления (Object) .....	72
Связывание этажей (Floor link) .....	73
Камера (Camera) .....	74

График (Graph) .....	74
Надпись (Text Label) .....	75
Иконки, используемые при создании визуализации (Visualization icons) .....	76
Утилиты (Utilities) .....	77
EnOcean .....	79
Предупредительные сообщения (Alerts) .....	80
Журнал (Logs) .....	81
Журнал ошибок (Error logs) .....	81
Справка .....	82
<b>WEBI USER MODE VISUALIZATION (УПРАВЛЕНИЕ ОБЪЕКТАМИ ВИЗУАЛИЗАЦИИ, НОРМАЛЬНЫЙ) .....</b>	<b>83</b>
<i>Touch visualization – режим графического представления данных с использованием сенсорных устройств</i> .....	83
<b>WEBI TOUCH VISUALIZATION (УПРАВЛЕНИЕ ОБЪЕКТАМИ ВИЗУАЛИЗАЦИИ МОБИЛЬНЫЙ) .....</b>	<b>84</b>
<i>Взаимодействие Modbus RTU/TCP с LM2</i> .....	84
<i>Master functions</i> .....	85
<i>Визуализация Modbus объектов</i> .....	86
<i>Пример использования (residentscript)</i> .....	86
<i>Взаимодействие со встроенными портами ввода-вывода</i> .....	89
Взаимодействие с последовательным портом RS232 .....	90
<b>WEBI USER MODE SHEDULERS (УПРАВЛЕНИЕ ЗАПЛАНИРОВАННЫМИ СОБЫТИЯМИ СИСТЕМЫ) .....</b>	<b>91</b>
<b>WEBI TREND LOG (ПРОСМОТР СТАНДАРТНЫХ ОТЧЁТОВ СИСТЕМЫ) .....</b>	<b>92</b>
<b>СТАНДАРТНЫЕ СХЕМЫ ПОДКЛЮЧЕНИЯ .....</b>	<b>93</b>
 ... 	

## Авторские права

Авторские права принадлежат компании Embedded Systems SIA© 2011.

Все права защищены.

## Уведомление

EVIKA сохраняет за собой право при необходимости вносить изменения в приведенную ниже информацию. Компания не несет ответственности за ошибки и неточности, допущенные в настоящем руководстве. Руководство предназначено исключительно для содействия операторам в использовании аппаратного и программного обеспечения программируемого контроллера Logic Machine 2.

## Товарные знаки

Товарный знак EVIKA принадлежит компании "ООО Эвика". Настоящим подтверждается, что все прочие наименования и товарные знаки являются собственностью их владельцев.

## Вводная часть

Контроллер Logic Machine 2 предназначен для реализации комплексной логики работы устройств в сетях KNX/EIB. Программируемый контроллер позволяет эффективно выстраивать процесс автоматизации зданий в соответствии с предъявляемыми требованиями, обеспечивая конечным пользователям неограниченные возможности при минимальных затратах.

Контроллер Logic Machine 2 представляет собой встраиваемую платформу с интерфейсами FT1.2 и Ethernet, что позволяет расширять возможности KNX/EIB, управляя логикой работы устройств сети и отношениями между объектами посредством пользовательских скриптов. Встроенный веб-сервер, набор готовых решений и шаблонов скриптов с открытым кодом обеспечивают удобный, пользовательско-ориентированный конфигурационный интерфейс. Используя пользовательские скрипты, Logic Machine 2 может исполнять функции, панели управления термостатирования, системой безопасности, освещением и т. п. одновременно.

Для создания скриптов используется язык программирования промышленных платформ - Lua, дополненный модулем поддержки KNX устройств. Следует отметить и ряд других отличительных характеристик Logic Machine 2: запуск функций и резидентных скриптов и как по расписанию, так и по событиям, автоматическое обнаружение объектов; встроенные часы реального времени.

Logic Machine 2 показывает прекрасные эксплуатационные качества в различных окружающих условиях, в том числе и сложных.

## Техническая поддержка

Неисправные устройства подлежат возврату в компанию EVIKA на территории РФ, СНГ и Embedded Systems на территории стран Евросоюза.

В случае возникновения каких-либо вопросов в части технического обслуживания устройства необходимо обратиться в службу технической поддержки. Часы работы: 9:00–17:00 (понедельник – пятница) GMT + 04:00. Контактная информация:

Адрес электронной почты: support@evika.ru

Тел.: + 7(495) 988-09-91

Обновления для встроенного программного обеспечения: [www.evika.ru](http://www.evika.ru)



### Предупреждения !:

#### ➤ Безопасность

Инсталляция электрического оборудования может производиться только квалифицированным электриком. Устройства не должны использоваться в приложениях, которое прямо или косвенно поддерживают безопасность и здоровье человека или животных, или для сохранности больших материальных ценностей.

#### ➤ Монтаж

Устройства поставляются в рабочем состоянии. Входящие в комплект поставки соединители используются по мере необходимости.

➤ **Электрические соединения**

Устройства разработаны для работы при безопасном низком напряжении (SELV).  
Заземление не требуется. Следует избегать скачков напряжения при переключениях питания.

# Терминология

---

## **Logic Machine 2**

Обобщённый контроллер-представитель семейства Logic Machine 2

- LM2-CON
- LM2-INT
- LM2-Re:actor

В местах где описываются специфические особенности, контроллер будет названо конкретно.

## **Устройство, Контроллер**

То же что и Logic Machine 2.

## **ПК, Персональный Компьютер.**

## **ОС, Операционная Система.**

## **Система**

Инженерная система, составной частью которой является Logic Machine 2.

## **Объект**

## **Инсталлятор**

Специалист создающий систему, в том числе подключающий и настраивающий Logic Machine 2 для работы в этой системе.

## **Пользователь**

Конечный потребитель, использующий функции системы, подготовленные для него Инсталлятором и производителем.

## **Скрипт**

Текст Программы для исполнения Logic Machine 2.

## **Хелпер**

Заранее подготовленный фрагмент Скрипта для выполнения какой либо задачи. Хелперы предназначены для использования в качестве шаблонов в программах инсталляторов.

## **WEB Сервер**

Это программно-аппаратный комплекс, размещенный внутри Logic Machine 2 предназначенный для реализации функций WEB Интерфейса.

## **WEB Интерфейс**

Это программа, исполняемая WEB Сервером, которая:

- Формирует страницы отображаемые в браузере;
- Принимает команды, запросы и ответы сделанные через браузер посетителем
- Читает или изменяет данные объектов системы, запускает программы по командам полученным из Браузера.

## **Браузер**

Стандартная Программа, инсталлированная на ПК посетителя для работы с WEB Сервером, просмотра формируемых им страниц и формирования запросов для него.

## **Посетитель**

Любой человек использующий WEB Интерфейсы Logic Machine 2

## **Селектор Интерфейсов, Станица Выбора Интерфейса**

Это первая страница отображаемая WEB Сервером, после входа на него посетителя по базовому адресу. Общий WEB Интерфейс в Logic Machine 2 разделён на несколько отдельных программ, специализирующихся по работе с определёнными функциями или посетителями.

Страница предназначена для выбора конкретного WEB Интерфейса. Подробнее см. раздел: раздел: Селектор Интерфейсов (стр. 18).

**WEBI\_\***

Некоторые WEB Интерфейсы на WEB Сервере не имеют коротких собственных названий. Префикс WEBI означает что имеется в виду WEB Интерфейс Logic Machine 2, \* конкретизирует его по разделу Селектор Интерфейсов (стр. 18).

**FlashSYS**

Это один из WEB интерфейсов имеющий собственное имя. Альтернативное название: WEBI Network configuration.

Подробнее см. раздел: раздел: Network configuration (FlashSYS) (стр. 20).

**Визуализация**

Программно аппаратные средства для графического представления объектов системы, их свойств и взаимосвязей, а также их коррекции.

**KNX, KNX/EIB**

Один из современных стандартов распределённого управления инженерным оборудованием, широко применяющийся для целей диспетчеризации и автоматизации зданий.

**ETS**

Программа на ПК Инсталлятора предназначенная для обслуживания и настройки сетей KNX.

<http://www.konnex-russia.ru/knx-standard/knx-tools/ets/>





# Описание устройства

## Обзор функций и возможностей

### Аппаратное обеспечение

➤ **На основе надежного и быстрого процессора ARM9 400MHz**

Индустриальные температурные режимы эксплуатации; аппаратные функции самоконтроля; защита от перенапряжений по питанию и LAN обеспечивает непрерывное и надежное функционирование. Допускается установка в жестких промышленных условиях. Не требует технического обслуживания.

➤ **Встроенный интерфейс FT1.2**

Настраивается через сеть Ethernet и содержит шлюз для сети KNX/EIB. Поэтому, дополнительный внешний интерфейс TPUART или FT1.2 не нужен, кроме того, через Logic Machine 2 можно настраивать и другие устройства сети. Возможен так же вариант удаленной настройки через Internet.

➤ **Встроенные часы реального времени**

Часы реального времени хранят точное время системы, что обеспечивает своевременный запуск скриптов. Часы обеспечены резервным питанием.

➤ **Встроенный самоконтроль.**

Logic Machine 2 содержит отдельный дополнительный процессор, предназначенный для контроля работы основного процессора и других аппаратных частей устройства.. Если он обнаружит ненормальное поведение, то попытается исправить эту ситуацию.

➤ **Низкие требования к питанию.**

Logic Machine 2 имеет очень низкую мощность потребления и низкие требования к питанию. Для подключения устройства в щитах можно использовать широкую номенклатуру источников, в том числе, используемых для питания KNX шины или отдельные малогабаритные источники. Реально, так же, обеспечить длительную автономную работу от блока резервного питания небольшой мощности.

### Программное обеспечение и скрипты

➤ **Настройка, диагностика и программирование через WEB Интерфейс.**

Logic Machine 2 имеет встроенный WEB сервер, реализованный на AJAX, содержащий все необходимые функции для настройки, программирования и диагностики устройства. Инсталлятор может получить доступ к WEB серверу через любой современный Browser, поэтому устанавливать дополнительное ПО не требуется. В случае присвоения устройству реального IP-адреса и его поддержке, обеспечивается возможность удаленного доступа через Internet.

➤ **Обновление фирменного ПО.**

Фирменное ПО Logic Machine 2 может быть произведено дистанционно, включая обновление самого WEB сервера.

➤ **Визуализация для планшетов Android/iOS**

➤ **Автоматическое обнаружение всех объектов**

Logic Machine 2 автоматически добавляет новые объекты в базу данных системы при получении от них телеграмм; необходимость ручной идентификации отсутствует. Новые объекты готовы для программирования и использования, после установки их типа данных.

➤ **Язык скриптов Lua**

Для создания скриптов Logic Machine 2 применяется высокоэффективный, облегченный и высокопроизводительный язык программирования Lua. Его расширенная библиотека функций позволяют не только обеспечить работу с объектами в рамках сети KNX, но и выходить за её границы, соединяясь с другими сетями и службами.

➤ **Хелперы и набор готовых решений.**

Редактор скриптов в WEB интерфейсе предлагает программисту возможность вставлять готовые кусочки кода из подготовленного и организованного набора, что существенно облегчает и

ускоряет процесс программирования. Набор поставляется производителем и может редактироваться пользователем.

Кроме того, имеется набор готовых решений из которых можно позаимствовать конкретные решения по подключению оборудования и соответствующие скрипты.

➤ **Имена объектов и метки**

Вы можете использовать уникальные имена или метки для обращения к Вашим объектам, не прибегая к использованию конкретных адресов сети. Это даёт возможность легко переносить и повторно использовать Ваши программы. Система меток позволяет одновременно работать со множеством похожих объектов одновременно.

➤ **Выполнение скриптов по событию и по расписанию. Резидентные скрипты**

Logic Machine 2 использует скрипты трех видов.

**Запускаемые по событию** – скрипты, выполняются в случае возникновения соответствующего события в шине. Обычно они применяются при необходимости работы системы в реальном времени.

**Запускаемые по расписанию** – скрипты, выполняемые в определенное время. Могут быть использованы при работе с различными системами безопасности и для имитации присутствия.

**Резидентные** – скрипты, используются для избирательного опроса изменения статуса объектов. Как правило, применяются в сочетании с системами вентиляции и отопления, когда данные принимаются от нескольких групповых адресов.



## Технические характеристики

### Семейство продуктов

Logic Machine 2 выпускается в следующих модификациях:

Артикул	Основные различия
LM2-CON	Базовая модель, корпус 4 DIN
LM2-INT	Модель с 3-мя портами RS232 (внедрены за счет других портов ), корпус 4 DIN Unit
LM2-Re:actor	Модель с расширенным набором портов ввода вывода корпус 6 DIN Unit Позволяет создавать системы автоматизации «All-in-One» объединяя в одном корпусе возможности свободно программируемого контроллера LogicMachine2 и максимум периферии. Может использоваться как одиночное устройство для небольших инсталляций, так и в составе больших систем автоматизации, делает инсталляции экономически эффективными и гибкими.

### Спецификация устройств

	Параметр	LM2-CON	LM2-INT	LM-ReActor <sup>2</sup>
Контроллер	Процессор	ARM9-based 400 MHz		
	Память оперативная	64 MB с произвольной выборкой		
	Память энергонезависимая (программы и переменные)	256 MB NOR		
	Часы реального времени	Резервная батарейка, автоматическая синхронизация с системным временем.		
Порты и подключения	<u>Порты</u>			
	FT1.2 (KNX Bus), (Стандартный соединитель KNX Black&Red)	1 x		
	Ethernet10 BaseT / 100 BaseTX, (RJ-45)	1 x		
	USB 2.0	2 x		1 x
	RS485 <sup>1</sup>	1 x		2 x
	<u>Выходы</u> <sup>1</sup>			
	Выход дискретный. Открытый коллектор.	4 x	1 x	16 x
	Выход аналоговый, 0 .. 10 V	-	-	2 x
	<u>Входы</u> <sup>1</sup>			
	Дискретный вход или вход резистивного датчика типа PT1000	-		6 x
	Аналоговые входы Программно могут интерпретироваться как дискретные.	0 .. 10 V		0 .. 30 V
		4 x	1 x	8 x
	Антенна EnOcean (SMA Type)	-		1 x
Протоколы	Поддержка	HTTP, SMTP, KNX, Modbus, BACnet, GSM, GSM/GPRS, EnOcean, DMX, Ekey, SIP, XML, RSS, JSON, XMPP Подробнее смотрите в разделе: Поддержка стандартов (стр. 12)		
Питание	Допустимое рабочее напряжение	12 .. 30 V, DC		
	Потребляемая мощность не более	1.3 W		2.0 W

	Параметр	LM2-CON	LM2-INT	LM-ReActor <sup>2</sup>
Управление локальное	Индикаторы	L1 - индикатор состояния загрузки CPU, L2- Активность		
	Кнопка	Reset, много функциональная, подробнее смотрите в разделе Кнопка Reset		
Эксплуатация	Температура Рабочая	-5...+55°C		
	Температура Хранения	-20...+70°C		
	Пыле-Влаго защита по DINEN 60529	IP20		
	Электро Защита	Разъемы Ethernet и питания защищены от статического электричества и перенапряжения		
Конструкция	Стандартный DIN (35) Корпус, единиц	3		6
	Габаритные размеры,	52x90x51 mm		105x90x51 mm
	Вес	0.1 kg	0.1 kg	0.19 kg
	Материал корпуса	Пластик (серый)		
Сертификация	Сертификаты CE	EMBS-CE-110926/01		
	Сертификаты EMS	EN61000-6-1 EN61000-6-3		
	Сертификат РосТест	РОСС LV.АГ88.В32424		
	Гарантийный срок	24 мес		

**Примечания:**

- 1 Соединители Cage Clamp 2.5 mm<sup>2</sup>.
- 2 Приведены данные Технического Задания. Серийный выпуск исполнения пока не начат. Возможны коррекции. Подробнее см. [www.openrb.com/logic-machine-reactor/](http://www.openrb.com/logic-machine-reactor/)

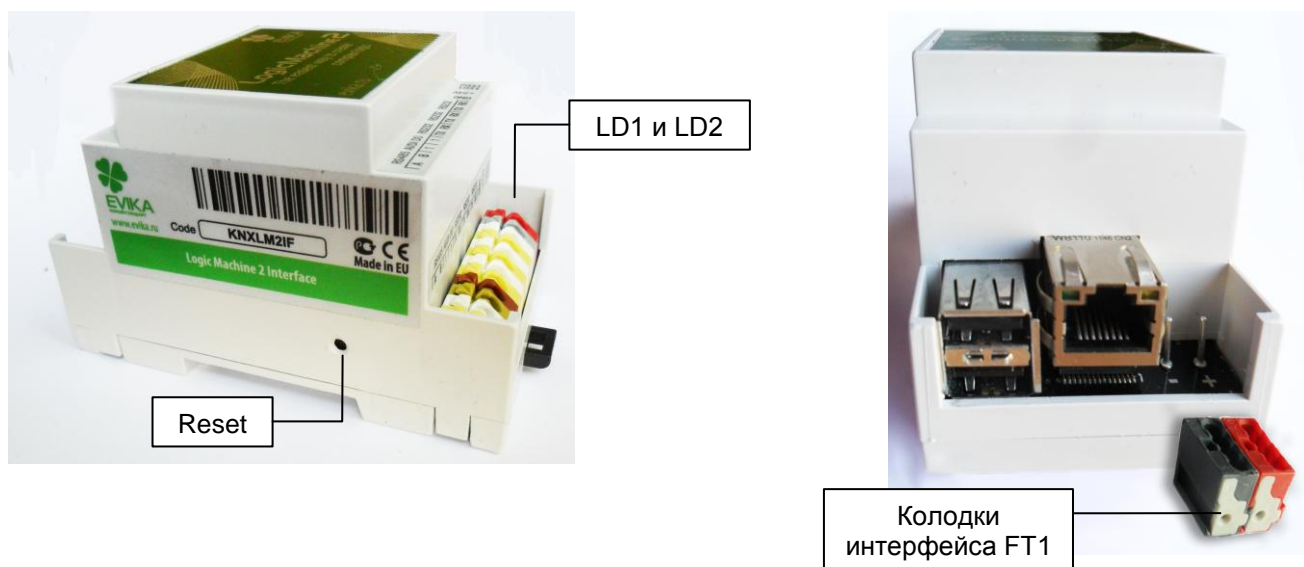
**Поддержка стандартов**

- HTTP
- SMTP, SSL;
- KNX/EIB TP, KNXnet/IP;
- Modbus TCP, Modbus RTU;
- BACnet IP, BACnet MS/TP;
- GSM (Huawei и другие USB модемы) для отправки SMS уведомлений и управления через SMS команды;
- EnOcean (для LM2-CON и LM2-INT через USB);
- DMX (через RS485);
- Ekey биометрические системы доступа (через RS485);
- XML (экспорт значений объектов, ошибок, сообщение о тревогах);
- RSS (ошибки, сообщение о тревогах);
- JSON, XMPP.

Система создана таким образом, чтобы каждый протокол мог взаимодействовать каждый с каждым, например BACnet – EnOcean или Modbus - GSM

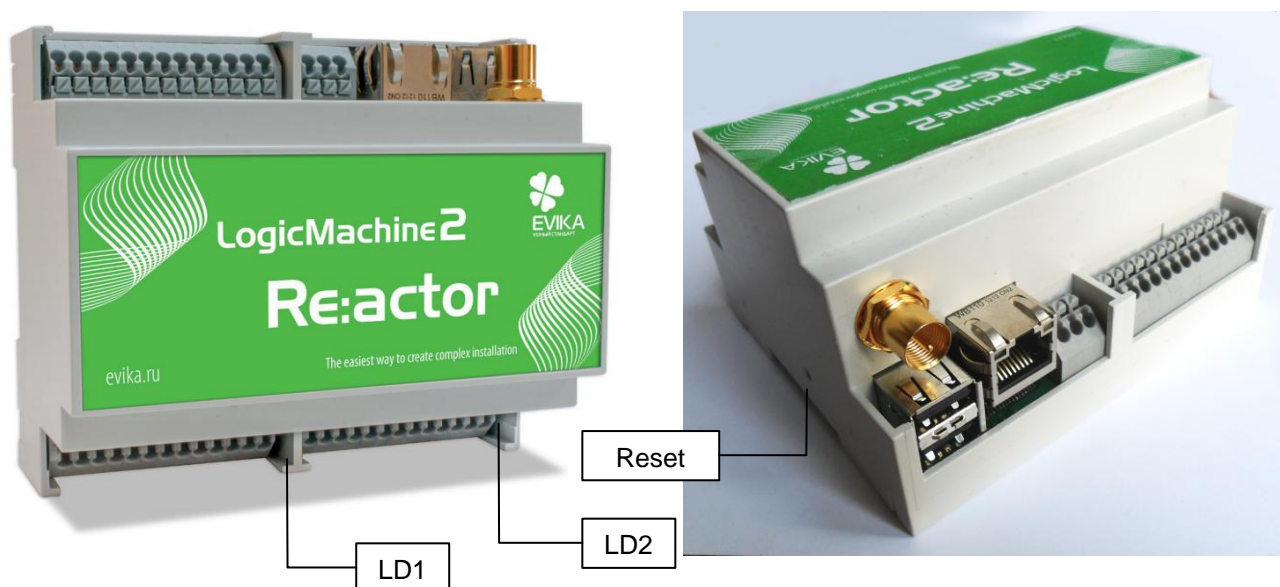
## Внешний вид контроллеров

### LM2-CON и LM2-INT



Расположение контактов показано в разделе: Стандартные схемы подключения (стр. 93)

### LM2-Re:actor



**Внимание!** исполнение LM2-Re:actor находится в разработке. Здесь показан опытный образец.

### Кнопка Reset \ FactoryReset

В случае нарушения нормального функционирования Logic Machine 2 (зависание), устройство можно перезапустить кнопкой Reset.

Отверстие кнопки Reset находится на боковой стороне корпуса:

- у LM2-Re:actor справа,
- у остальных слева.

Нажатие осуществляется стержнем диаметром не более 2-ух мм.

Кнопка выполняет следующие функции:

Функция	Удержание Кнопки
<b>Reset (Перезагрузка)</b> Инициализация устройства как при включении питания. При нормальном функционировании устройства может быть выполнена программно. Подробнее в разделе: FlashSys\System\Reboot (Перезагрузка) (стр. 24)	менее 10 sec
<b>Factory Reset (Восстановление заводских настроек)</b> Восстановление заводских значений параметров устройства, Подробнее в разделе: Заводские значения параметров (стр. 14) Программы пользователя не стираются. В завершении операции производится Перезагрузка.	более 10 sec



## Индикаторы LED1 и LED2

### Индикатор LED1 - Статус центрального процессора.

Не горит	Неисправность или отсутствие питания.
Мигает	Нормальная работа, Чем чаще, тем больше загрузка.
Горит	Процессор перегружен.

### Индикатор LED2 - Статус устройства.

Не горит	Неисправность или отсутствие питания.
Мигает, 1 s	Нормальная работа.
Горит	Находится в режиме ShutDown. Подробнее в разделе: FlashSys\System\Shutdown (Подготовка к отключению питания) (стр. 24)

## Комплект поставки

Оборудование	LM2-CON или LM2-INT	LM2-Re:actor
Контроллер, с заводскими установками готовый к настройке и программированию скриптов.	1	1
Упаковка Картонная	1	1
Блок питания 24 V (для "DeskTop" использования (настройка, проверка) контроллера, тип не уточняется)	1	1
Колодки шины KNX TP1 (FT1.2 Red&Black)	1 пара	-
Антенна EnOcean выносная для LM2-Re:actor.	-	1

## Заводские значения параметров

### Установки паролей посетителей

Тип пользователя	Параметр	Значение
admin	Login:	admin
	Password:	admin
remote	Login:	remote
	Password:	remote

Тип пользователя	Параметр	Значение
visadmin	Login:	visadmin
	Password:	visadmin
viscontrol	Login:	viscontrol
	Password:	viscontrol
visview	Login:	visview
	Password:	visview

Подробнее в разделе: FlashSys\System\GUI login (Назначение паролей доступа) (стр. 22)

#### Установка подключения Ethernet

Параметр	Значение
Интерфейсный порт	Ethernet
Протокол связи (Protocol)	Static_IP
IP-адрес страницы выбора интерфейса	192.168.000.010
Сетевая маска для LAN	255.255.255.000
Gateway IP	
DNS1	

Подробнее в разделе: Параметры подключения (стр. 27)

#### Установка подключения BACnet

Параметр	Значение
Device Id	
Password	

Подробнее в разделе: Параметры подключения (стр. 27)

#### Установка интерфейсных линий

Параметр	Значение
Режим интерфейсных линий для LM2-Re:actor	Вход

#### Программа системного мониторинга

Восстанавливается из системной директории,

Подробнее в разделе: FlashSys\Services\System Monitoring (Скрипт системного мониторинга) (стр. 31)

Выше перечисленные значения параметров устанавливаются также при процедуре Factory Reset, Подробнее в разделе: Кнопка Reset \ FactoryReset (стр. 13)





# Установка

---

**Предупреждение!:**

Перед подключением Logic Machine 2 к шине KNX, желательно задать настройки для всех объектов KNX.

## Включение и отключение питания

Рекомендуемый порядок коммутации питания:

- Порядок включения: Питание Logic Machine 2, Питание KNX.
- Порядок отключения обратный.





# Программирование и настройка

---

Logic Machine 2 настраивается и программируется только через встроенный в него WEB интерфейс.

Отлаживать программы автономно (без подключения к целевому Logic Machine 2), нельзя. При необходимости автономной отладки, следует использовать другое устройство Logic Machine 2, подключенное к компьютеру программиста.

**Внимание!** Следует учитывать возможные различия релизов системного ПО загруженных на отладочной и рабочей Logic Machine 2.

WEB\_Interface предоставляет все необходимые средства для импорта/экспорта программ и данных устройства, включая случаи подключения через несколько сетей (удаленно).

## Совместимость с браузерами

WEB\_Interface прошёл тестирование на:

- Google Chrome
- Apple Safari
- Mozilla Firefox

## Подключение к WEB интерфейсу

### Вы подключены к сети устройства

Если ПК и устройство находятся в одной сети, просто введите его IP Адрес в адресную строку браузера. Описание настроек можно посмотреть в разделе Параметры подключения (стр. 27).

Эти настройки Вам следует сохранить отдельно (распечатав или сохранив в файле ПК).

Если эти настройки утеряны, то можно попытаться воспользоваться приложениями поиска IP адресов:

- Apple Discovery app.
- Сервисом zeroconf
- Средствами контроля подключенных сетевых устройств (например через интерфейсы DHCP роутеров).
- и т.п.

Если определить параметры подключения не удалось, то устройство можно привести к заводским настройкам (Кнопка Reset \ FactoryReset (стр. 13)).

### Устройство подключено к другой сети

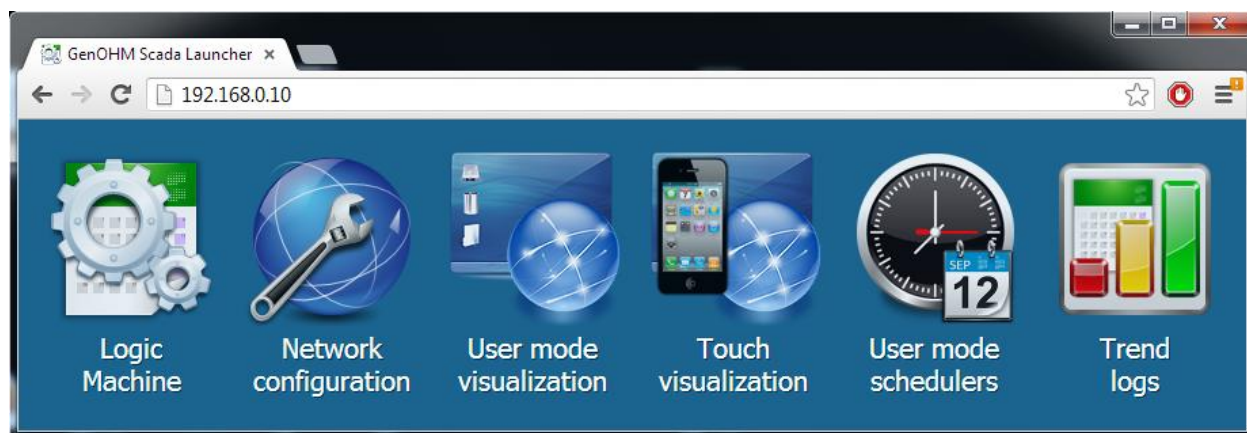
Для доступа к устройству необходимо назначить IP адрес, выделенный провайдером канала связи, роутеру, и настроить роутер для трансляции передаваемых данных по этому адресу на локальный IP адрес устройства.

Локальный адрес определяется как описано выше.

Подробнее см. FlashSys\Networks\Interfaces (Доступные сетевые подключения) (стр. 25).

## Селектор Интерфейсов

После удачного подключения WEBI отобразит страницу выбора интерфейса.



### Интерфейсы Инсталлятора:

➤ **Logic Machine (Интерфейс инсталлятора)**

- Скрипты,
- Редактирование визуализаций,
- Настройка объектов и параметров KNX,
- Связи между объектами.

Подробнее см. раздел: WEBI Logic Machine (Интерфейс инсталлятора) (стр. 36)

➤ **Network configuration (Интерфейс администратора устройства, сетей и подключений)**

- настройка сетей IP и KNXnet/IP, BackNet,
- Выгрузка и Загрузка пользовательских данных.,
- Обновление ПО,
- Перезагрузка,
- и т.д.

Подробнее см. раздел: Network configuration (FlashSYS) (стр. 20).

### Интерфейсы Пользователей:

➤ **User mode visualization (Управление объектами визуализации, нормальный)**

- Пользовательский интерфейс, через компьютеры с большим монитором.

Подробнее см. раздел: WEBI User mode visualization (Управление объектами визуализации, нормальный) (стр. 83).

➤ **Touch visualization (Управление объектами визуализации мобильный)**

- Пользовательский интерфейс, через мобильные устройства и сенсорные экраны (iPhone/iPod/iPad итп).

Подробнее см. раздел: WEBI Touch visualization (Управление объектами визуализации мобильный) (стр. 83).

➤ **User mode Shedulers (Управление запланированными событиями системы)**

- Пользовательский интерфейс
- Интерфейс находится в стадии тестирования

➤ **Trend log (Просмотр стандартных отчётов системы)**

- Пользовательский интерфейс: статистика и протоколы событий системы.
- Интерфейс находится в стадии тестирования.

Некоторые функций для удобства инсталляторов дублируются в нескольких интерфейсах.



## Сервер данных.

Logic Machine 2 содержит ещё один WEB интерфейс, под названием Remote. Но он не графического типа.

Remote предназначен для приема автоматических стандартных запросов от машин пользователей и возврата им пакетов данных, содержащих статистику работы системы или накопленные системой данные, например показания коммунальных счётчиков для служб ЖКХ.



## Права доступа к интерфейсам и функциям

Настройки устройства защищаются технологией авторизации пользователей системы по Имени типа пользователя и его паролю.

### Типы Пользователей

В системе поддерживаются пользователи 5-ти типов:

Имя типа пользователя	Все права	Server Статистики	Настройка	Этаж	Этаж закрытый	Объекта	Объект закрытый
admin	RW	R	A	RW	RW	RW	RW
remote	-	R	A	-	-	-	-
visadmin	-	-	A	RW	R	RW	RW
viscontrol	-	-	A	R	R	RW	R
visview	-	-	-	R	R	R	R

Где

#### Поля:

Все права	Даётся право на доступа и изменение любых объектов системы, в том числе прав установки паролей остальных типов пользователей.
Server Данных	Дает доступ к отправке запроса на получение пакета данных от устройства. Подробнее см. раздел: Сервер данных (стр. 19).
Настройка	Даёт право входа в интерфейсы WEBI Logic Machine и FlashSYS. Право входа в остальные графические интерфейсы есть у всех пользователей. Подробнее см. раздел: Селектор Интерфейсов (стр. 18).
Этаж	Даёт право доступа к объекту: Этаж незакрытый <sup>1</sup> .
Этаж Закрытый	Даёт право доступа ко всем объектам типа Этаж, в том числе закрытым <sup>1</sup> .
Объект	Даёт право доступа к незакрытым <sup>2</sup> Объектам на незакрытых этажах <sup>1</sup> .
Объект Закрытый	Даёт право доступа ко всем объектам, на незакрытых этажах <sup>1</sup> .

#### Права:

-	нет прав ни на чтение ни на запись, ни на вход.
R	есть право на чтение
W	есть право на запись
A	есть право Входа в интерфейсы WEBI-LM2 и FlashSYS.

Конкретных пользователей в системе нет.

Одновременно в системе могут работать несколько обезличенных пользователей одного и того же или различных типов. При конфликте доступа к объектам на запись, система запомнит данные связанные с последним запросом от любого пользователя (с учетом прав).



# Network configuration (FlashSYS)

Программа WEBI Network configuration имеет собственное имя - FlashSYS. Она является общей для различных устройств EVIKA, управляемых по сетям TCP/IP.

Далее будет описана работа системы с релизом FlashSYS v2 r90 (07.02.13)

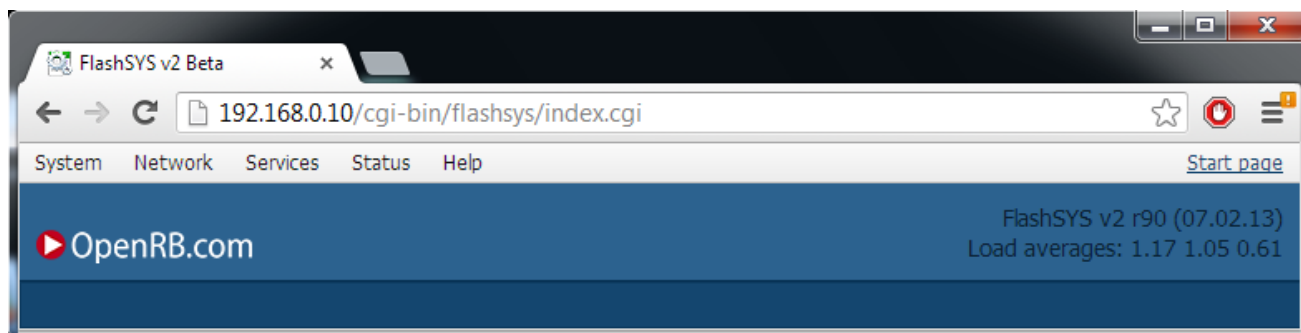
## Назначение FlashSYS

В интерфейсе FlashSYS инсталлятор может:

- Настроить управление роутером Logic Machine 2,
- Настроить доступ к функциям системы,
- Обновить программное обеспечение,
- Провести перезагрузку,
- Настроить шлюз в KNX шину,
- Узнать статусы работы системы и сетевых интерфейсов.
- и многое другое.

## Начальная страница FlashSys

О подключении и выборе интерфейса FlashSys см. раздел: Подключение к WEB интерфейсу (стр. 17).



## Меню FlashSys

System	Системные настройки и программы. Подробнее в разделе: FlashSys\System (Системные настройки и программы) (стр. 21).
Network	Это основной параметр КУ, подробности его определения мы рассмотрим в разделе: FlashSys\Networks (стр. 24).
Services	Это основной параметр КУ, подробности его определения мы рассмотрим в разделе: FlashSys\Services (Сетевые сервисы) (стр. 31).
Status	Это основной параметр КУ, подробности его определения мы рассмотрим в разделе: FlashSys>Status (Системная статистика и мониторинг) (стр. 33).
Help	Это основной параметр КУ, подробности его определения мы рассмотрим в разделе: FlashSys>Status\Help (Помощь) (стр. 35).
Start page	Переход на страницу Селектора Интерфейсов



## Релиз FlashSys

FlashSys v2 r90	Обозначение релиза программы FlashSys, загруженной ("прошитой") в данный момент в устройстве).
-----------------	--

## Load Average статус (Загрузка центрального процессора)

Load Average	Показывает текущую загрузку центрального процессора - первая группа цифр.
--------------	---

## FlashSys\System (Системные настройки и программы)

Предлагаются следующие диалоги:

Hostname	Имя WEB сервера Logic Machine 2 прд этим именем Logic Machine 2 можно будет увидеть в различных приложениях отображающих имена подключенных IP устройств сети.
Packages	Просмотр и управление библиотеками программ и данных Logic Machine 2 Подробнее см. на стр. 21.
GUI login	Установка паролей доступа в WEB Интерфейсы Подробнее см. на стр. 22
Backup and restore	Создание резервных копий параметров FlashSys в памяти Logic Machine 2, их загрузка и выгрузка на ПК инсталлятора Подробнее см. на стр. 22
Upgrade firmware	Обновление ПО устройства. Подробнее см. на стр. 23
Reboot	Перезагрузка Logic Machine 2 Подробнее см. на стр. 24
Shutdown	Подготовка к отключению питания Подробнее см. на стр. 24



## FlashSys\System\Packages (Библиотеки программ и данных)

Таблица показывает текущие загруженные и зарегистрированные библиотеки.

Поля:

### Package Name

Имя библиотеки

### Version

релиз библиотеки

### Кнопки



удалить библиотеку



Загрузить или обновить библиотеку

Packages		
Package name	Version	
avahi-daemon	0.6.30-2	
base-files	43.33-r30646	
busybox	1.15.3-3.4	
dropbear	0.53.1-5	
eibd	0.0.5	
flashsys2	85	
genohm-scada	20120419	
haserl	0.8.0-2	
Actions:		

Обновление библиотек производится также и в автоматическом режиме см. раздел: FlashSys\System\Upgrade firmware (Загрузка заводского ПО) (стр. 23).



## FlashSys\System\GUI login (Назначение паролей доступа)

Диалог для каждого типа пользователя устанавливает пароль на доступность функции или интерфейса.

Описание прав доступа см. раздел: Права доступа к интерфейсам и функциям (на стр. 19).

Вкладки диалога:

- |               |  |
|---------------|--|
| Admin, Remote | для WEBI Инсталляторов и Сервера Данных  |
| Visualization | параметры доступа для WEBI Пользователей |

Значения параметров нового устройства или после процедуре Factory Reset см. в разделе: Заводские значения параметров (на стр. 14).



## FlashSys\System\Backup and restore (Резервные копии параметров FlashSys)

Все текущие настройки FlashSys можно сохранить в специальной области памяти Logic Machine 2 в виде файла. Файлы настроек могут передаваться между Logic Machine 2 и ПК инсталлятора.

## Формат имени файла создаваемый Logic Machine 2:

`File_prefixYYYY_MM_DD.tar.gz`

Где

File_prefix	Часть имени файла задаваемая инсталлятором Может быть пустым
YYYY_MM_DD	Дата Создания файла по времени WEB сервера
.tar.gz	Стандартное окончание имени файла.



Создать и сохранить файл текущих параметров в памяти устройства. Программа спросит File\_prefix. Если полное имя файла совпадёт с имеющимся, старый будет перезаписан без предупреждения



Заменить текущие параметры параметрами из файла



Удалить файл



Скачать на PC инсталлятора



Загрузить на Logic Machine 2 файл с ПК инсталлятора, ограничения на символы имени остаются, но структуру имени кроме окончания можно изменить.



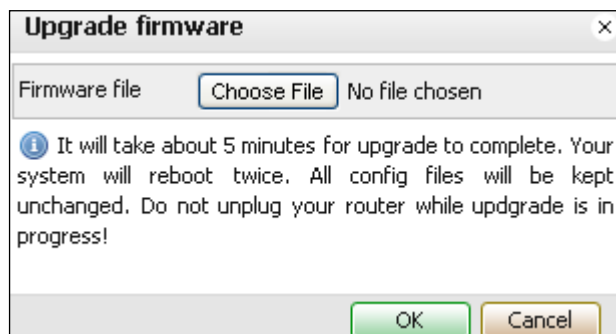
## FlashSys\System\Upgrade firmware (Загрузка заводского ПО)

Файл загрузки имеет расширение .img . Он содержит все необходимые пакеты программ и данных для обновления.

В настоящий момент файлы обновлений распространяются только через службу технической поддержки.

### Процедура загрузки.

- Получите файл или дистрибутив релиза заводского ПО.
- Убедитесь, что он подходит для Вашей ситуации (если будут какие то особенности, наши инженеры предупредят Вас об этом, или сопроводят дистрибутив обновления соответствующими указаниями).
- Если дистрибутив был запакован - распакуйте его.



**Внимание!** При загрузка firmware стираются программы и данные Инсталлятора.

- Сохраните на ПК Ваши программы и данные. Все параметры FlashSys сохраняются, однако рекомендуем также перенести на ПК и их резервную копию (если Вы до сих пор этого не сделали !).
- Войдите в диалог загрузки ПО.
- Нажмите кнопку Choose File (Выбрать Файл).
- В стандартном диалоге ОС ПК укажите полученный файл для загрузки.
- Ok
- Программа начнёт загрузку файла обновления на контроллер. Ваша ОС ПК может сопровождать этот процесс индикацией элемента типа Progress\_Bar (отображение времени до завершения операции)
- После удачного получения файла контроллер отобразит сообщение о начале записи полученного файла. С этого момента контроллер будет недоступен. Браузер может отобразить страницу, сообщающую о потере связи.



**Внимание!** С момента завершения загрузки файла и до момента окончания второй перезагрузки недопустимо:

- использование кнопки Reset,
- прерывание питания контроллера или выход параметров питающего напряжения за рамки нормального.

**Внимание!** После окончания заправки мы рекомендуем очищать кеш Вашего браузера.

- Через минуту, начинайте пробовать возобновлять связь с контроллером (обновить страницу).
- Вам следует начать беспокоиться, если через 5 минут после завершения загрузки файла в контроллер, система не сможет отобразить Селектор Интерфейсов.
- Восстановите Ваши программы и данные см.

## Определение текущих релизов установленного заводского ПО

В настоящий момент единого обозначения установленного релиза firmware нет. После записи в память Logic Machine 2 можно посмотреть текущие релизы у отдельных частей:

FlashSys см. раздел: Релиз FlashSys (на стр. 20).

WEBI Logic Machine см. раздел: Селектор Интерфейсов (на стр. 18).



### FlashSys\System\Reboot (Перезагрузка).

Перед началом перезагрузки системы Контролер попросит подтвердить запрос на перезагрузку.

Контроллер не оповещает о окончании процесса перезагрузки, Восстановление связи с контроллером будет возможно только после ручного обновления странице браузера или при новом подключении к WEB Сервер.

Во время перезагрузки Контролер производит ряд сервисных операций на своей файловой системе. Если там не возникает проблем, то перезагрузка занимает не более 3-х минут.

Контроллер можно перегрузить аппаратно, см. раздел: FlashSys\System\Reboot (Перезагрузка) (на стр. 24)



### FlashSys\System\Shutdown (Подготовка к отключению питания)

Для предотвращения повреждения файловой системы и потери важных текущих данных связанных с прерыванием питания на Logic Machine 2 можно запустить процедуру Shutdown.


Программа попросит подтвердить операцию, и в случае подтверждения начнёт необходимые действия.

**Внимание!** Выйти из состояния Shutdown в настоящее время можно только сбросом по питанию (кнопка Reset реализована программно).



## FlashSys\Networks

Interfaces	Сетевые интерфейсы Logic Machine 2 см.: FlashSys\Networks\Interfaces (Доступные сетевые подключения) (стр. 25).
Routes	Преобразование сетевых адресов. см.: FlashSys\Network\Routes (Таблица маршрутизации) (стр. 28).
KNX connection	Управление подключением к KNX шине. см.: FlashSys\Network\KNX connection (параметры KNXIP роутера) (стр. 29).
ARP table	Перечень подключенных IP устройств. см.: FlashSys\System (Системные настройки и программы) (стр. 21).
BACKnet	Управление подключением к сети BACnet. см.: FlashSys\Network\BACnet (подключение к сети BACnet) (стр. 31).

**Внимание!** С целью возможности согласованного изменения подключений, настройки этого раздела запоминаются во временном буфере. При наличии изменений, в правой части строки меню интерфейса появляется кнопка . После нажатия на кнопку, значения из буфера переписываются в постоянную память и производится соответствующая переналадка подключений.



Если от изменений нужно отказаться,  
перезагрузите FlashSYS: Start Page > Network Configuration.

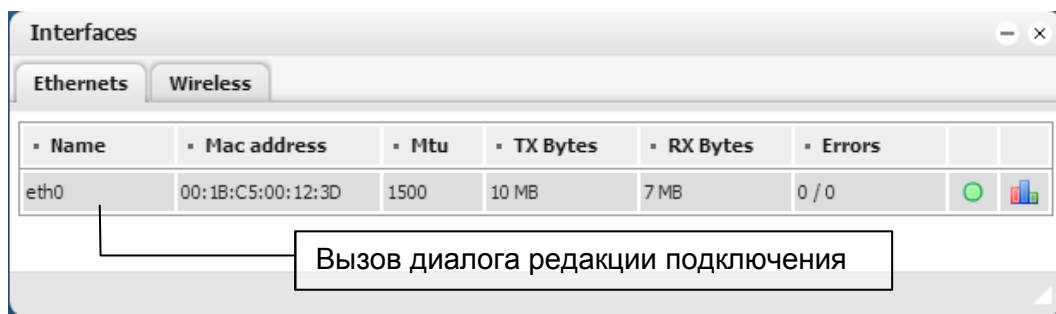


## FlashSys\Networks\Interfaces (Доступные сетевые подключения)

Диалог содержит 2 вкладки

Ethernet	Проводные Подключения Logic Machine 2 Подробнее см. на стр. 27
Wireless	БезПроводные подключения Logic Machine 2 Подробнее см. на стр. 26.

### FlashSys\Networks\Interfaces\Ethernet (Ethernet подключения)



Вкладка отображает доступные проводные подключения Ethernet:

#### Поля

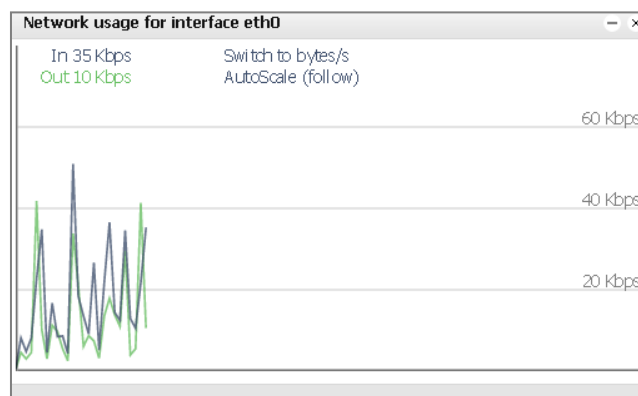
Name	Имя типа подключения Logic Machine 2 к сети не редактируется, по умолчанию eth0. Клик по значению поля приведёт к отображению диалога редакции параметров подключения. Подробнее в разделе: Параметры подключения (стр. 27).
Mac address	Уникальный адрес MAC сетевого Ethernet устройства. совпадает с адресом прошитым изготовителем элементов интерфейса. Не редактируется.
MTU	Максимальный размер пакета передаваемых данных.
TX Bytes	Количество переданной информации с момента перезагрузки.
RX Bytes	Количество принятой информации с момента перезагрузки.
Errors	количество пакетов с ошибками переданных и принятых соответственно.
	Кнопка включения/выключения и индикатор статуса подключения. Зеленый цвет элемента означает подключенное состояние, красный - отключенное.

**Внимание!** Если данное подключение к ПК инсталлятора единственное, то клик по зелёной кнопке приведёт потере связи с устройством, восстановить которую будет возможно только произведя аппаратную перезагрузку (кнопка Reset или переключение питания).

**Внимание!** Действие выполняется немедленно, без ожидания подтверждения Apply changes, см. стр. 24



Кнопка вызова диалога статистики подключения и загрузки соединения. , где в режиме реального времени выстраивается график нагрузки на интерфейс (ввод/вывод). Доступно две единицы измерения: кбайт/сек. и байт/сек.



## **FlashSys\Networks\Interfaces\Wireless (Беспроводные подключения)**

Вкладка отображает доступные беспроводные подключения (WiFi).

Структура вкладки и работа с ней - аналогично вкладке Ethernet (стр. 25).

## Параметры подключения Ethernet

Диалог вызывается кликом по соответствующему значению поля Name в диалоге Параметры подключения (стр. 25).

Значения заводских Параметров см. в разделе Заводские значения параметров (стр. 14).

Состав полей диалога зависит от текущего значения поля Protocol

### Protocol

протокол подключения:

- Static IP     статический IP адрес контроллера;
- DHCP        IP адрес контроллера назначается автоматически.
- PPPoE        адрес контроллера назначается автоматически по протоколу PPP.

#### Static IP

The screenshot shows the 'Interface eth0' dialog box with the 'General' tab selected. The 'Protocol' dropdown is set to 'Static IP'. The fields are filled with the following values: IP address: 192.168.1.211, Network mask: 255.255.255.0, Gateway IP: 192.168.1.1, DNS server: 192.168.1.1, and Mtu: (empty). The 'OK' and 'Cancel' buttons are at the bottom right.

#### IP address

заданный статический IP-адрес.

#### Network mask

сетевая маска

#### Gateway IP

IP-адрес шлюза по умолчанию. При наличии данного параметра нет необходимости дублировать значение в окне Routes.

#### DNS Server

адрес сервера DNS.

#### MTU

максимальный размер пакета, передаваемого по протоколу обмена данными.

#### DHCP

The screenshot shows the 'Interface eth0' dialog box with the 'General' tab selected. The 'Protocol' dropdown is set to 'DHCP'. The fields are filled with the following values: Current IP: 192.168.1.211, DNS server: 192.168.1.1, and Mtu: (empty). The 'OK' and 'Cancel' buttons are at the bottom right.

#### Current IP

IP-адрес, присвоенный сервером DHCP автоматически.

#### DNS Server

адрес сервера DNS.

#### MTU

максимальный размер пакета, передаваемого по протоколу обмена данными.

## Настройки туннеля PPPoE

### Username

имя пользователя для соединения с сервером PPPoE.

### Password

пароль.

### Keep alive

время активного подключения.

### Dial on Demand

подключение по требованию.

### Radius server

адрес сервера RADIUS для авторизации.

### DNS Server

адрес сервера DNS.

### MTU

максимальный размер пакета, передаваемого по протоколу обмена данными.

Interface eth0

General

Protocol: PPPoE

Username:

Password:

Keepalive:

Dial on Demand:

Radius server:

DNS server: 192.168.1.1

Mtu:

OK Cancel



## FlashSys\Network\Routes (Таблица маршрутизации)

### Вкладка Dynamic (динамические маршруты)

#### Interface

название интерфейса.

#### Destination

IP-адрес назначения.

#### Gateway

IP-адрес шлюза.

#### Network mask

сетевая маска.

#### Flags

Interface	Destination	Gateway	Network mask	Flags
ath0	192.168.2.0	*	255.255.255.0	U
eth0	192.168.1.0	*	255.255.255.0	U
eth0	224.0.0.0	*	224.0.0.0	U
eth0	default	192.168.1.1	0.0.0.0	UG

### Вкладка Static (статические маршруты)

Содержание вкладки аналогично вкладке Dynamic (динамические маршруты)



## FlashSys\Network\KNX connection (параметры KNXIP роутера)

В этом диалоге настраиваются параметры Logic Machine 2 исполняющего стандартные функции KNXIP роутера (устройства соединяющего 2 сети или подсети).

### Вкладка General

#### Mode

Тип интерфейса KNX у  
Logic Machine 2

- FT1.2 (по умолчанию)
- TP-UART
- EIBnet IP Tunneling

#### Parameter

Последовательный объект  
OpenWRT отвечающий за  
соединение с KNX.  
/dev/ttyS1 - пока единственное  
значение.

#### KNX address

физический адрес Logic Machine 2  
как устройства в KNX.

#### KNX IP features

Разрешить работу с KNX интерфейсом.

#### MulticastIP

Общий адрес для передачи.

#### Multicastinterface

групповой интерфейс. Используется для отправки телеграмм KNX в другие сети KNX через протокол TCP/IP (см. раздел: FlashSys\Networks\Interfaces (Доступные сетевые подключения) (на стр. 25)).

#### Maximum Telegrams in Quene

максимально количество телеграмм в очереди на обработку в буфере Logic Machine 2 (для предотвращения перегрузки процессора).

### Вкладка SRC filter

Контроллер может блокировать передачу проходящих через него телеграмм KNX шины по признаку определённых групповых адресов в её поле адрес отправителя.

#### SRC policy

Действие контролера по блокировке телеграммы:

- No filter  
не проверяет, пропускает всё.
- Accept selected individual addresses  
пропускает телеграммы только от адресов указанных в поле Address list.
- Drop selected individual addresses  
блокирует телеграммы от адресов указанных в поле Address list.

#### Address list

Перечень групповых адресов источников в телеграммах для фильтрации.

В каждой строке указывается один адрес. Можно использовать маску {\*}, например:

1.1.\*  
1/1/\*

**Внимание!** Для использования режима фильтрации включить KNX IP features (стр. 29).

## Вкладка DST group filter

Контроллер может блокировать передачу проходящих через него телеграмм KNX шины по признаку определённых групповых адресов в её поле адрес назначения.

### SRC policy

Действие контроллера по блокировке телеграммы:

- No filter  
не проверяет, пропускает всё.
- Accept selected individual addresses  
пропускает телеграммы только для адресов указанных в поле Address list.
- Drop selected individual addresses  
блокирует телеграммы для адресов указанных в поле Address list.

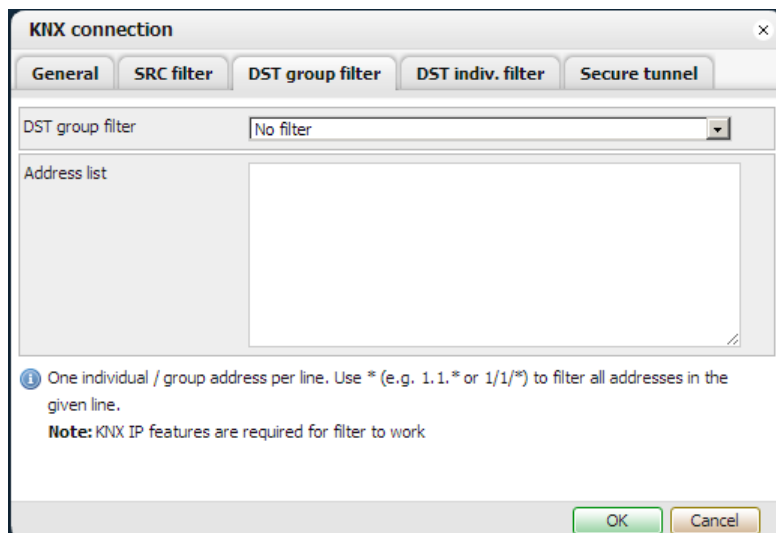
### Address list

Перечень групповых адресов получателей в телеграммах для фильтрации.

В каждой строке указывается один адрес. Можно использовать маску {\*}, например:

1.1.\*  
1/1/\*

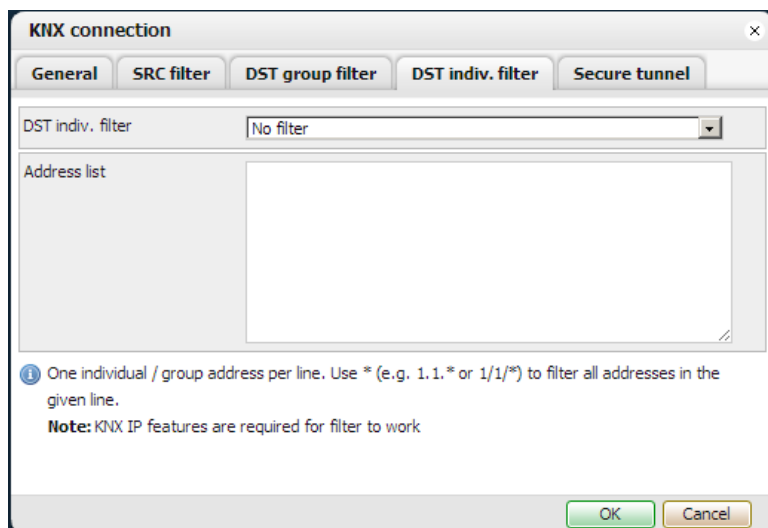
**Внимание!** Для использования режима фильтрации включить KNX IP features (стр. 29).



## Вкладка DST indiv. filter

Контроллер может блокировать передачу проходящих через него телеграмм KNX шины по признаку определённых ИНДИВИДУАЛЬНЫХ адресов в её поле адрес назначения.

Работа со вкладкой полностью аналогична работе с вкладкой DST group filter, но для физических адресов шины, (см. стр. 30)



## Вкладка Secure tunnel

Logic Machine 2 может организовать защищённый "туннель" между двумя узлами KNX сетей по протоколу TCP, который обеспечивает функцию подтверждения доставки пакета получателю.

### Secure tunnel

Роль для режима защищенного туннеля -- Disabled

- туннель не используется

- Client

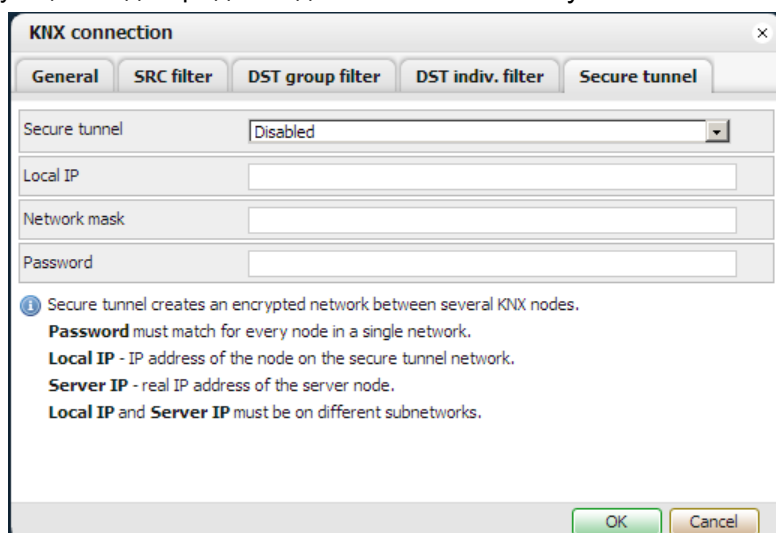
Logic Machine 2 выступает в роли Клиента.

- Server

Logic Machine 2 выступает в роли Сервера.

### Server IP

указать IP сервера при выборе режима Secure tunnel=Client.



**Local IP**

локальный IP адрес Logic Machine 2.

Server IP и Local IP должны принадлежать разным подсетям.

**Network mask**

сетевая маска.

**Password**

пароль.

**FlashSys\Network\BACnet (подключение к сети BACnet)**

Device Id и Password - соответствующие параметры подключения к сети BACnet

**Restart Server (Перезагрузка сервера BACnet)**

При выбранной опции будет послан запрос Reinitialize Device, который перезагрузит все объекты в BACnet.

**Внимание!** При перезагрузке сервиса BACnet, список приоритетов будет установлен в NULL.

BACnet	
Device ID	127001
Password	mybacpwd
Restart server	<input type="checkbox"/>
OK Cancel	

**FlashSys\Services (Сетевые сервисы)**

Сетевые сервисы позволяют организовать быстрый обмен файлами с Logic Machine 2 и настроить параметры внутреннего мониторинга.

**FlashSys\Services\FTP Server (Сервер FTP)**

FTP сервис позволяет напрямую обмениваться файлами между файловой системой Logic Machine 2 и ПК инсталлятора.

**Server status**

разрешение работы сервера FTP

- Enabled

FTP разрешён

- Disabled

FTP запрещён.

**Port**

порт для FTP обмена

**Username**

имя пользователя канала FTP

**Password**

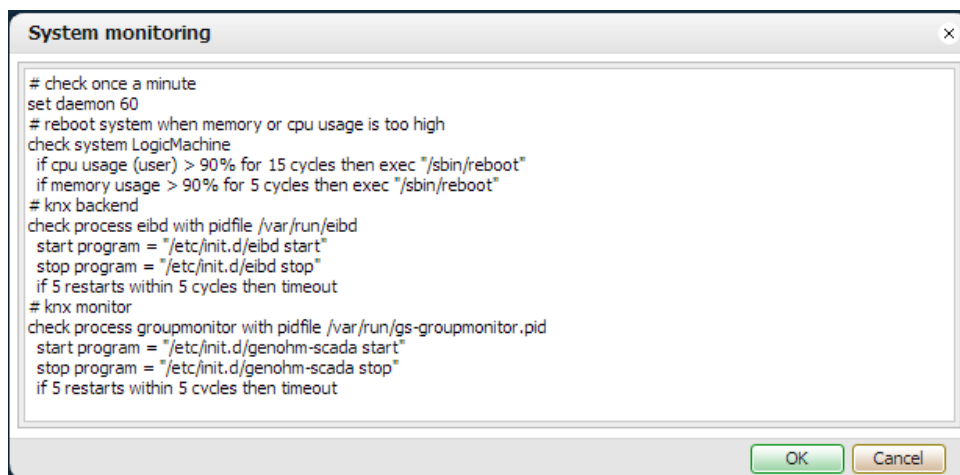
Пароль пользователя канала сервера FTP.

Если при сохранении значений диалога поле останется пустым, то его значение не меняется

FTP server	
Server status	Disabled
Port	21
Username	ftp
Password	
Leave password to blank to keep it unchanged	
OK Cancel	

**FlashSys\Services\System Monitoring (Скрипт системного мониторинга)**

в этом пункте открывается текстовый редактор программы, позволяющей определить поведение Logic Machine 2 при обнаружении критических ситуаций в активных процессах, подключенных сетях, загрузке центрального процессора и памяти.



Изменения вступают в силу после перезагрузки.

Файл восстанавливается до заводского при процедуре Factory reset, см. раздел: Заводские значения параметров (на стр. 14))

#### Возможные команды:

##### `set daemon NN`

Устанавливает период 1 цикла мониторинга.

NN Период цикла, s.

##### `check system LogicMachine`

определяет блок проверки загруженности центрального процессора Logic Machine 2.

##### `if cpu usage (user) RelOp Val% for CC cycles then Action`

определяет прведение Logic Machine 2 при перегрузке центрального процессора.

user	Пользователь
RelOp	Знак операции отношения ({=},{>},{<},{<=},{>=}).
Val	значение загрузки процессора в % для сравнения.
CC	Количество последовательных циклов выполнения указанного условия RelOp Val%, после которого управление передаётся оператору за словом then.
Action	оператор, исполняющийся в случае выполнения условия анализа загрузки центрального процессора.

##### `if memory usage RelOp Val% for CC cycles then Action`

определяет прведение Logic Machine 2 при перегрузке основной памяти.

RelOp	аналогично cpu usage.
Val	аналогично cpu usage.
CC	аналогично cpu usage.
Action	аналогично cpu usage.

##### `check process PName with pidfile PidFileName`

определяет блок самоконтроля процесса в процессоре Logic Machine 2.

PName	Имя Процесса.
PidFileName	Имя файловой переменной для сохранения данных мониторинга процесса.

##### `start program = PString`

##### `stop program = PString`

определяет запуск тестирующих процессов.

Синтаксис системного мониторинга так же расписан в заводском файле системного мониторинга в комментариях.





## FlashSys\Status (Системная статистика и мониторинг)

Сетевые сервисы позволяющие просмотреть накопленную статистику работы системы Logic Machine 2 и её текущее состояние, а так же выполнить некоторые системные действия.

System Status	Текущая информация о центральном процессоре, памяти и файловой системе см.: FlashSys\Status\System Status (Монитор системных ресурсов) (стр. 33).
Network Status	Текущая информация о подключениях к Logic Machine 2 см.: FlashSys\Status\Network Status (Статус подключений) (стр. 34).
Network Utilites	Утилиты проверки подключений Ping и Traceroute см.: FlashSys\Status\Network Utilites (Сетевые сервисные программы) (стр. 34).
System Log	Просмотр файла протокола работы системы Logic Machine 2 см.: FlashSys\Status\System Log (Протокол работы ОС) (стр. 35).
Running processes	Просмотр активных процессов центрального процессора, с возможностью их удаления см.: FlashSys\Status\Running processes (Монитор процессов) (стр. 35).



## FlashSys\Status\System Status (Монитор системных ресурсов)

Текущая информация о центральном процессоре, состоянии памяти и файловой системе.

### Вкладка General (Центральный процессор)

#### CPU model

наименование изготовителя и модели центрального процессора.

#### CPU BogomMips

Производительность центрального процессора

#### Linux kernel version

релиз ОС.

#### System uptime

продолжительность работы системы со времени последней перезагрузки в формате: Дней часов минут.

#### Load Averages

Загрузка центрального процессора  
подробнее см.: Load Average статус (Загрузка центрального процессора) (стр. 20)

Parameter	Value
CPU model	ARM926EJ-S rev 5 (v5I)
CPU BogomMips	199.06
Linux kernel version	3.8.2
System uptime	3d 7h 52m
Load averages	0.30 0.40 0.37

### Вкладка Memory usage (Основная память)

#### Total system memory

Объём основной памяти Logic Machine 2.

#### Used

Объём занятой памяти.

#### Free

Объём свободной памяти.

#### Buffered

Объём буфера.

#### Cached

Объём памяти занятой под Кэш буфер центрального процессора.

Memory info	Used
Total system memory	62568 kB
Used	24544 kB (39.23%)
Free	38024 kB (60.77%)
Buffered	0 kB
Cached	14576 kB

### Вкладка Partitions (Файловая система)

#### File System

Наименование раздела файловой системы

#### Total Size

Размер раздела.

#### Used

Объём занятой части раздела.

#### Free

Объём свободной части раздела.

File system	Total size	Used	Free	Mounted on
rootfs	250.0M	12.6M (5%)	237.4M	/
/dev/root	250.0M	12.6M (5%)	237.4M	/
tmpfs	30.6M	1.4M (5%)	29.1M	/tmp
tmpfs	512.0K	0 (0%)	512.0K	/dev

**Mounted on**

Ссылка на процесс-владельца раздела.

**FlashSys\Status\Network Status (Статус подключений)**

Общие параметры сети, IP-настройки и информация о переданных данных.

Network status							
Name	Mac address	IP address	Mtu	TX Bytes	RX Bytes	Errors	
ath0	00:80:48:52:D3:BB	192.168.2.211	1500	0 B	0 B	0 / 0	
eth0	00:50:C2:1E:AF:FE	192.168.1.211	1500	2 MB	4 MB	0 / 0	
eth1	00:50:C2:1E:AF:FD		1500	0 B	0 B	0 / 0	

Диалог аналогичен диалогу из FlashSys\Networks\Interfaces\Ethernet (Ethernet подключения) (стр. 25). Дополнительно выводится:

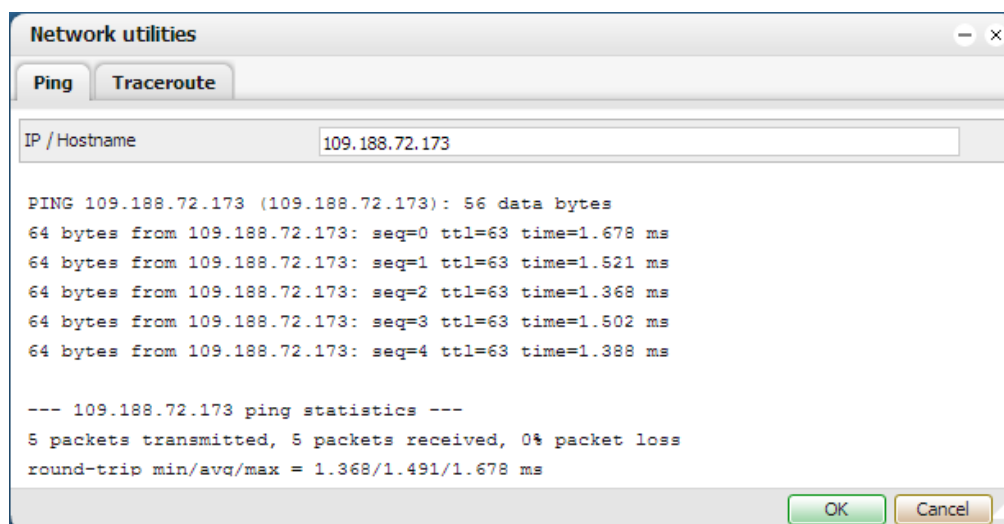
IP address      Текущий IP адрес подключения.

**FlashSys\Status\Network Utilites (Сетевые сервисные программы)****Вкладка Ping**

Программа позволяющая проверить наличие определённого узла сети и оценить качество связи с ним.

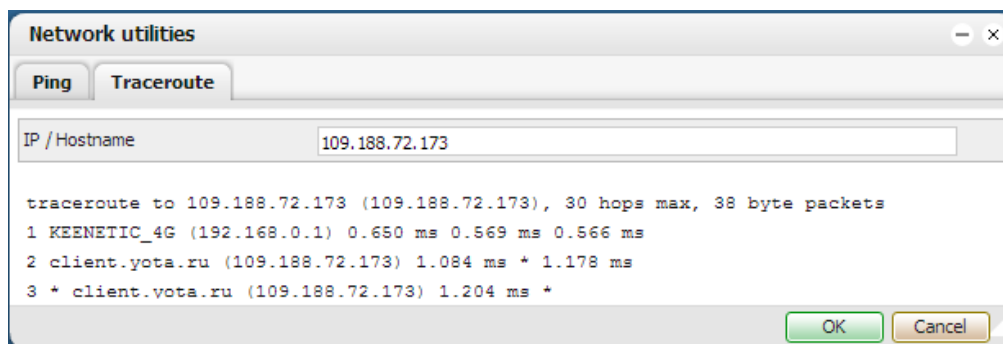
Обратится к тестируемому узлу можно по

- IP
- или имени узла HostName (см. FlashSys\System (Системные настройки и программы) (стр. 21)), если этот сервис правильно настроен и доступен.

**Вкладка Traceroute**

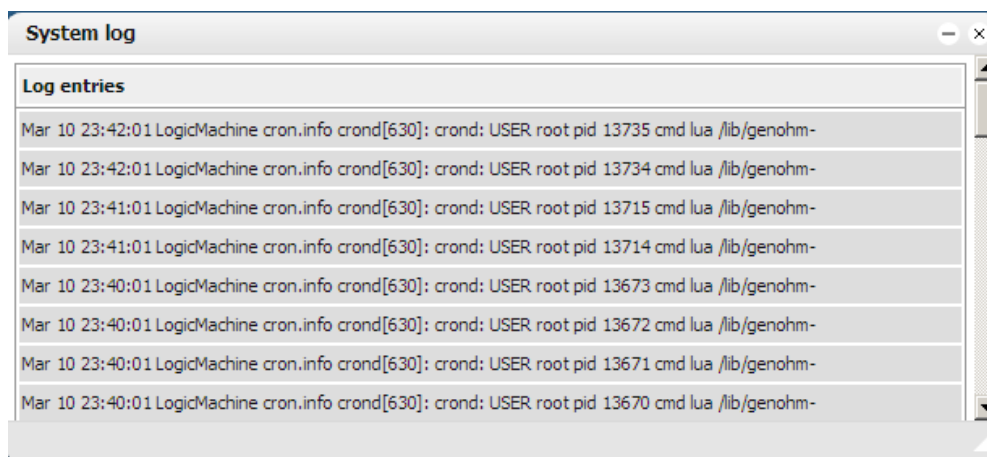
Программа позволяющая проследить прохождение тестовых пакетов по узлам сетей.

Выбор тестируемого узла аналогично Ping.



## FlashSys\Status\System Log (Протокол работы ОС)

В отчёте отражены события системы ОС



- Время
- и Другие параметры событий.



## FlashSys\Status\Running processes (Монитор процессов)

В диалоге отображаются все активные процессы ОС

### PID (Номер процесса)

Идентификационный номер процесса.

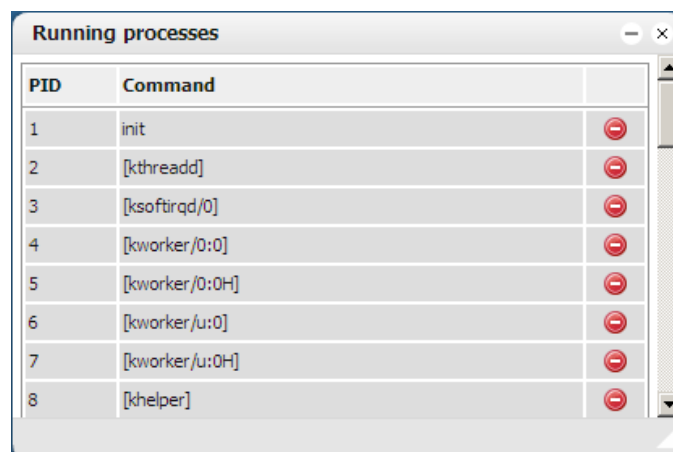
### Command (Комманда запуска)

Комманда запустившая процесс



### (Удалить процесс)

Удаляет процесс из очереди.



## FlashSys\Status\Help (Помощь)

пока в этом разделе находится только диалог About со ссылками на производителя.

Работы по созданию Online помощи ведутся.

# WEBI Logic Machine (Интерфейс инсталлятора)

---

Далее будет описана работа системы с Релизами:  
WEBI 20130225

## Общая концепция программирования Logic Machine 2

### ➤ Преимущества

- Простота и скорость создания стабильно работающей системы.
- Интуитивно понятные действия.
- Применение языка программирования, а не ограниченного набора опций.
- Возможность усложнения функциональности, за счет применения расширенных функций языка программирования.
- Возможность создавать свои "примитивы" и использовать их в дальнейшем как шаблоны.
- Интеграция с другими системами с использованием библиотек производителя.
- Для программирования и настройки достаточно средств уже заложенных в интерфейсы устройства.
- Возможность дистанционного программирования, отладки, настройки и мониторинга устройства, а в некоторых случаях и всей системы.
- Быстрая реакция разработчиков на запросы клиентов и
- Возможность обновления системы.

### Технология.

- Разработка проекта.
- Подбор оборудования системы.
- Монтаж системы.
- Назначение прав доступа инсталляторам и пользователям.
- Назначение адресов KNX.
- Первичное связывание объектов в рамках сети KNX (возможно и без контроллера)
- Создание логической структуры объекта, подготовка визуализационных картинок
  - Здания
  - Этажи
  - Помещения
  - Отдельные объекты
- Размещение иконок управления на визуализациях
- Связывание иконок с конкретными свойствами и функциями системы.
- Программирование скриптов функционирования иконок.
- Программирование скриптов взаимосвязи объектов и подсистем.
- Автоматическое построение представления системы для мобильных устройств
- Ручная подстройка автоматического представления.

Выше упомянутые пункты плана не являются обязательными. Вы можете начинать выполнение проекта с любого момента, корректируя выполненные работы по мере необходимости.

 ...

## Создание «поэтажной» структуры и добавления объектов на план

Подключитесь к Logic Machine с параметрами по умолчанию (**IP:** 192.168.0.10;  
**login/password:** admin/admin)

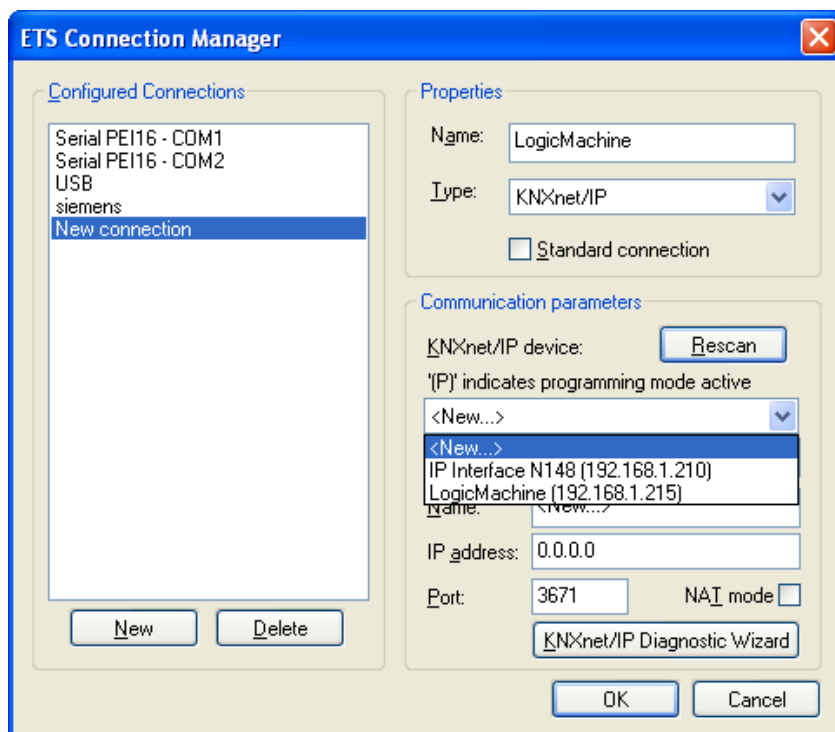
## Создание структуры “здание/этаж” и добавление объектов на план

Во вкладке Building задается структура здания, определяются этажи, загружаются фоновые изображения. Для добавления нового здания нажмите кнопку “*Add new building*”.

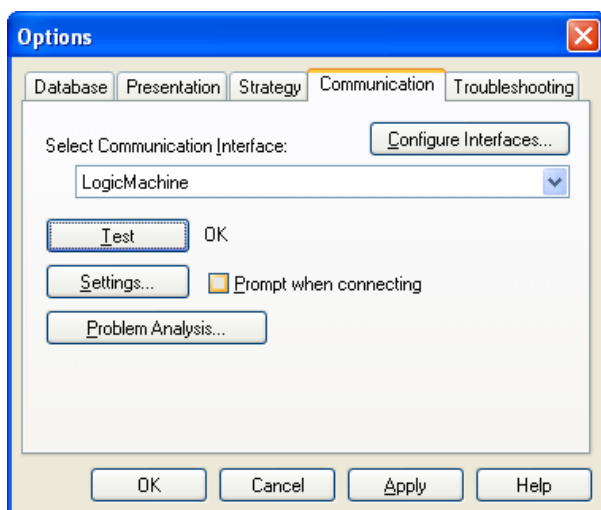
### Logic Machine 2 для настройки конфигурации сети KNX/EIB через протокол TCP/IP

Для использования Logic Machine 2 в рамках KNXnet/IP следует добавить устройство в *ETS Connection Manager*.

- Выбрать *Extras* → *Options* → *Communication* → *Configure interfaces*.



- Задать *Name* для соединения (по выбору).
- Выбрать *Type* = *KNXnet/IP*.
- Нажать кнопку *Rescan*; из выпадающего меню выбрать обнаруженный программируемый контроллер.
- Нажать *OK*.
- В окне *Options* → *Communication* выбрать вновь созданное соединение; в выпадающем меню установить его в качестве *Communication Interface*.
- Для проверки соединения при помощи ETS нажать кнопку *Test*.

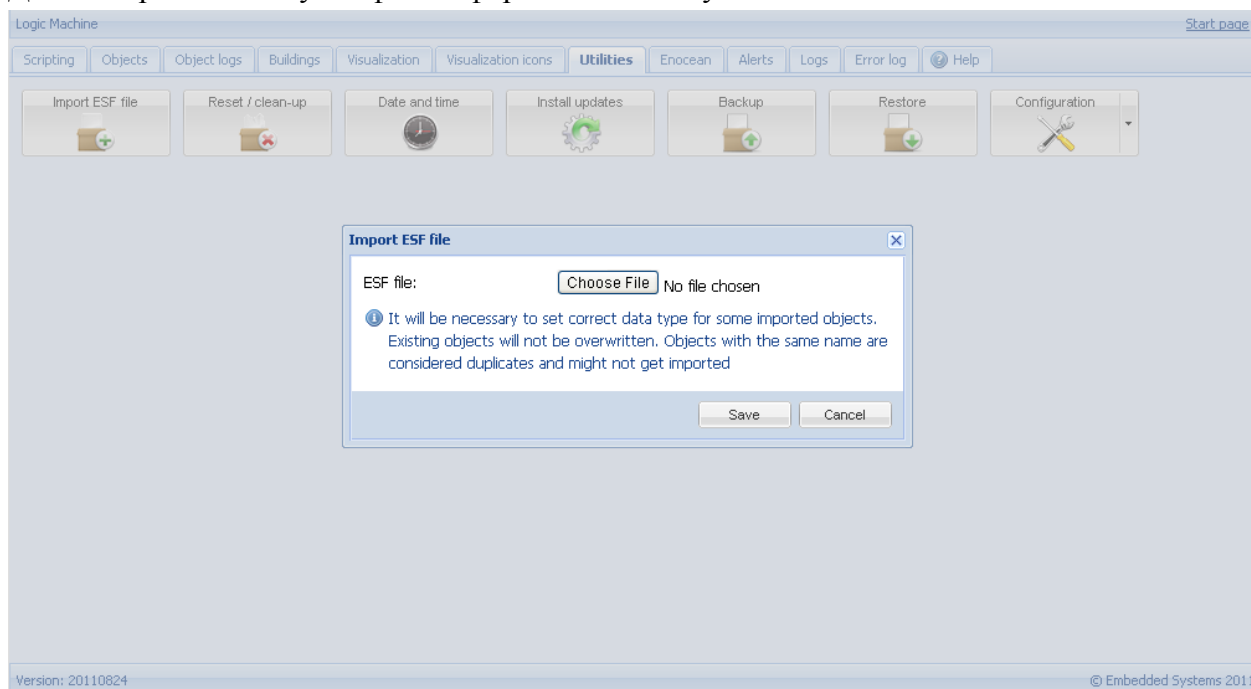


- Убедиться, что шине присвоен статус Online – в ETS нажать кнопку .

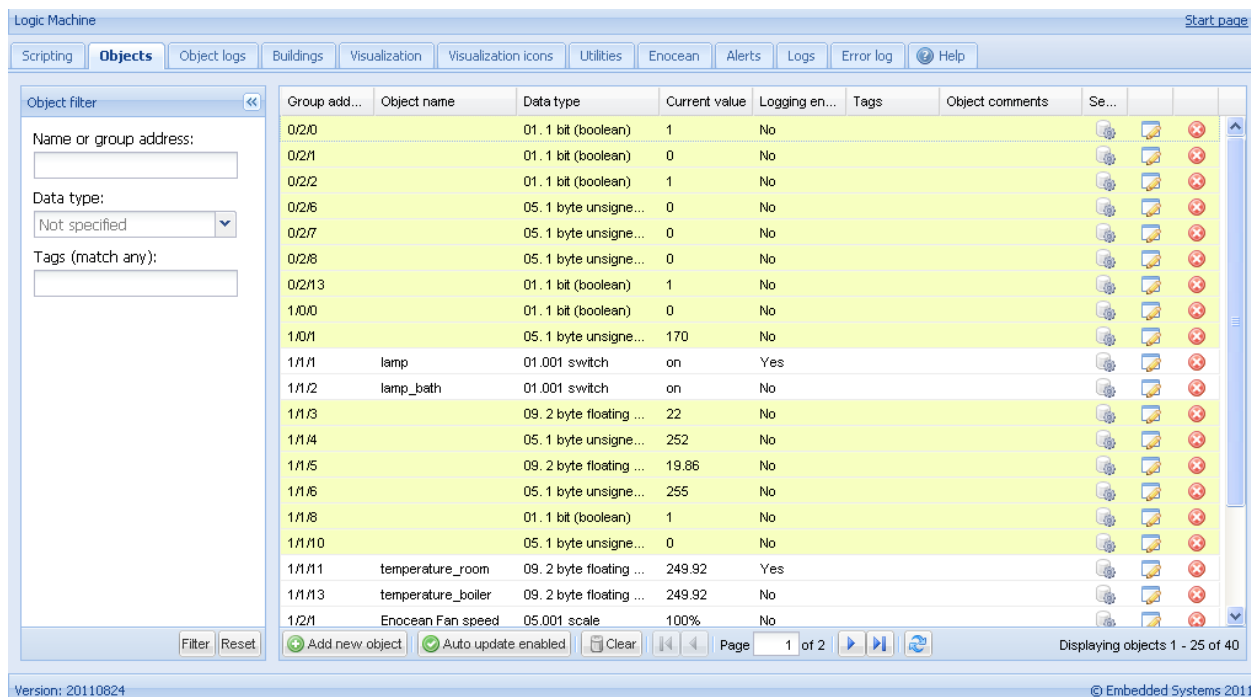
# Быстрый запуск. Создание визуализации для iPad/PC

## Импорт объектов

Для импорта используйте файлы формата ESF получаемые из ETS..

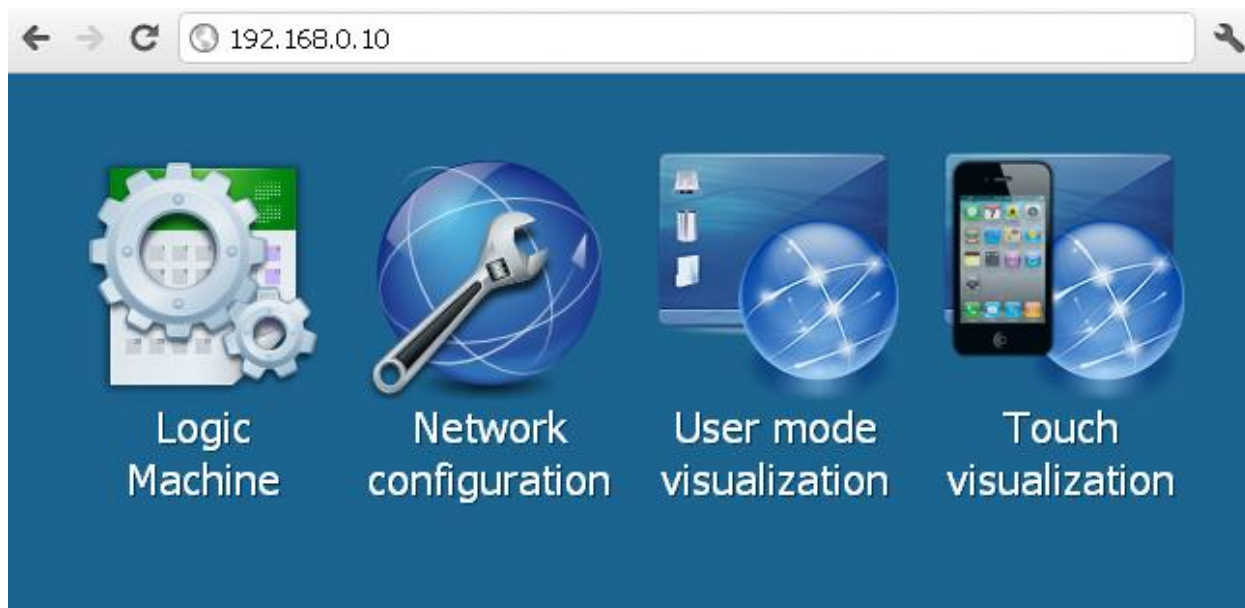


.. или подключите ЛМ к KNX сети и тогда объекты будут автоматически появляться во вкладке *Objects* когда они будут посылать телеграммы. Также можно добавить объекты вручную.



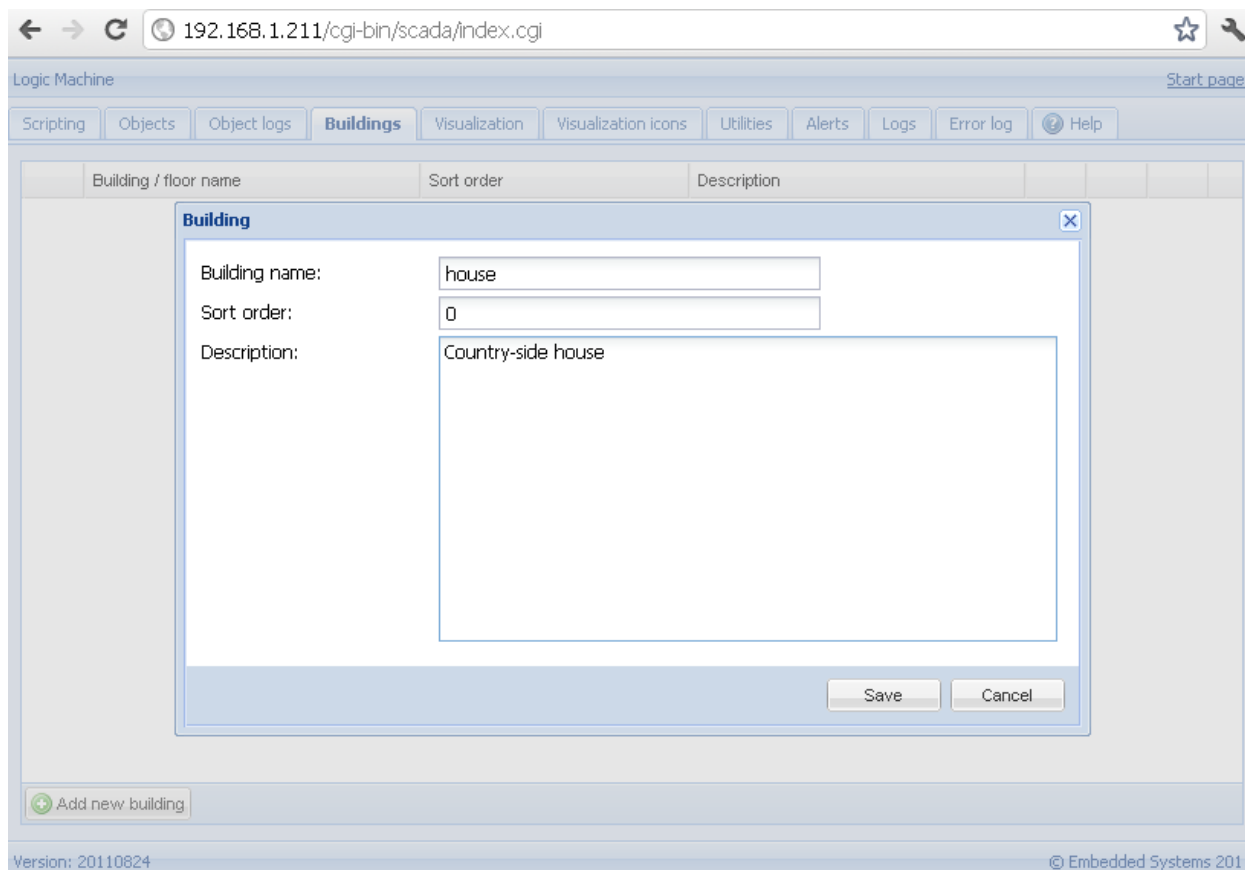
## Создание «поэтажной» структуры и добавления объектов на план

Подключитесь к Logic Machine с параметрами по умолчанию (**IP**: 192.168.0.10;  
**login/password**: admin/admin)





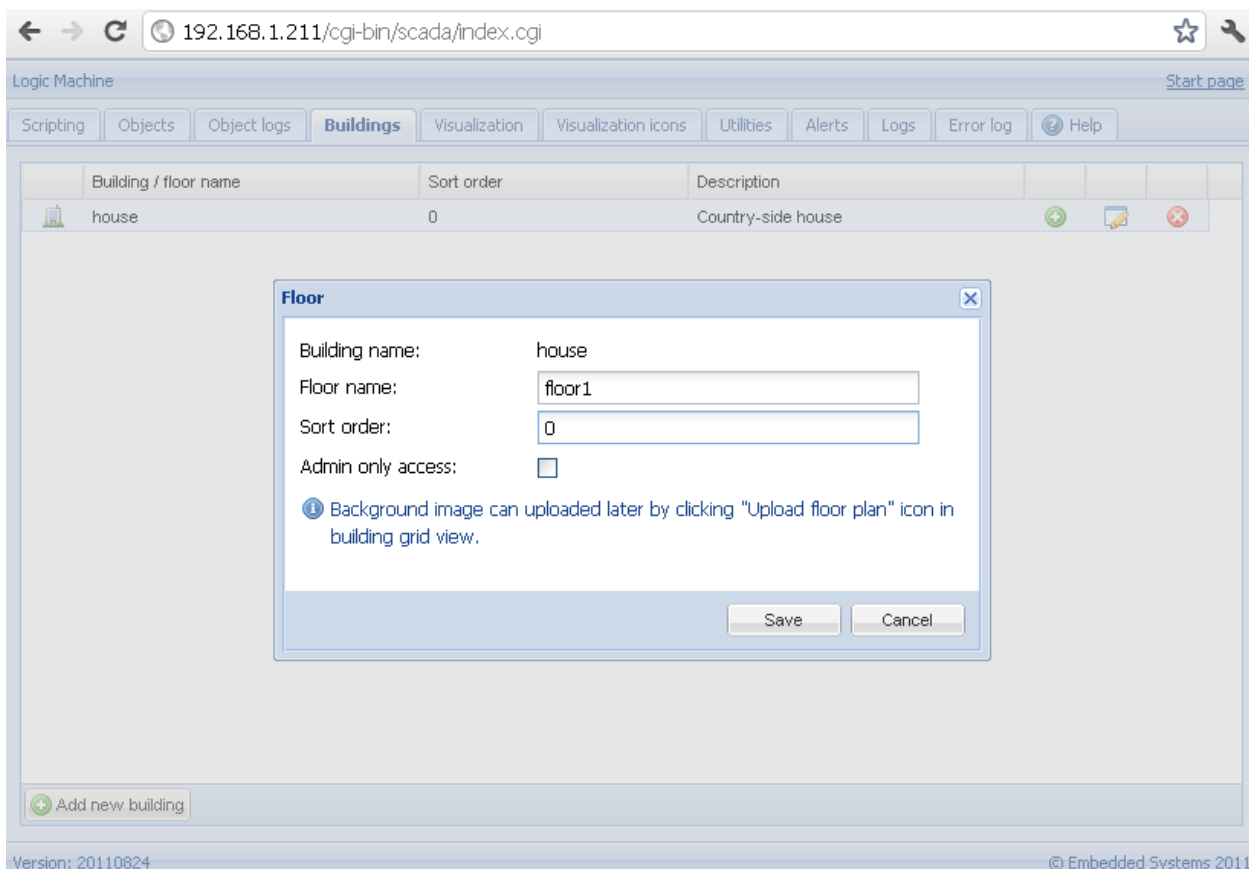
## Создание структуры “здание/этаж” и добавление объектов на план


Во вкладке Building задается структура здания, определяются этажи, загружаются фоновые изображения. Для добавления нового здания нажмите кнопку “Add new building”.

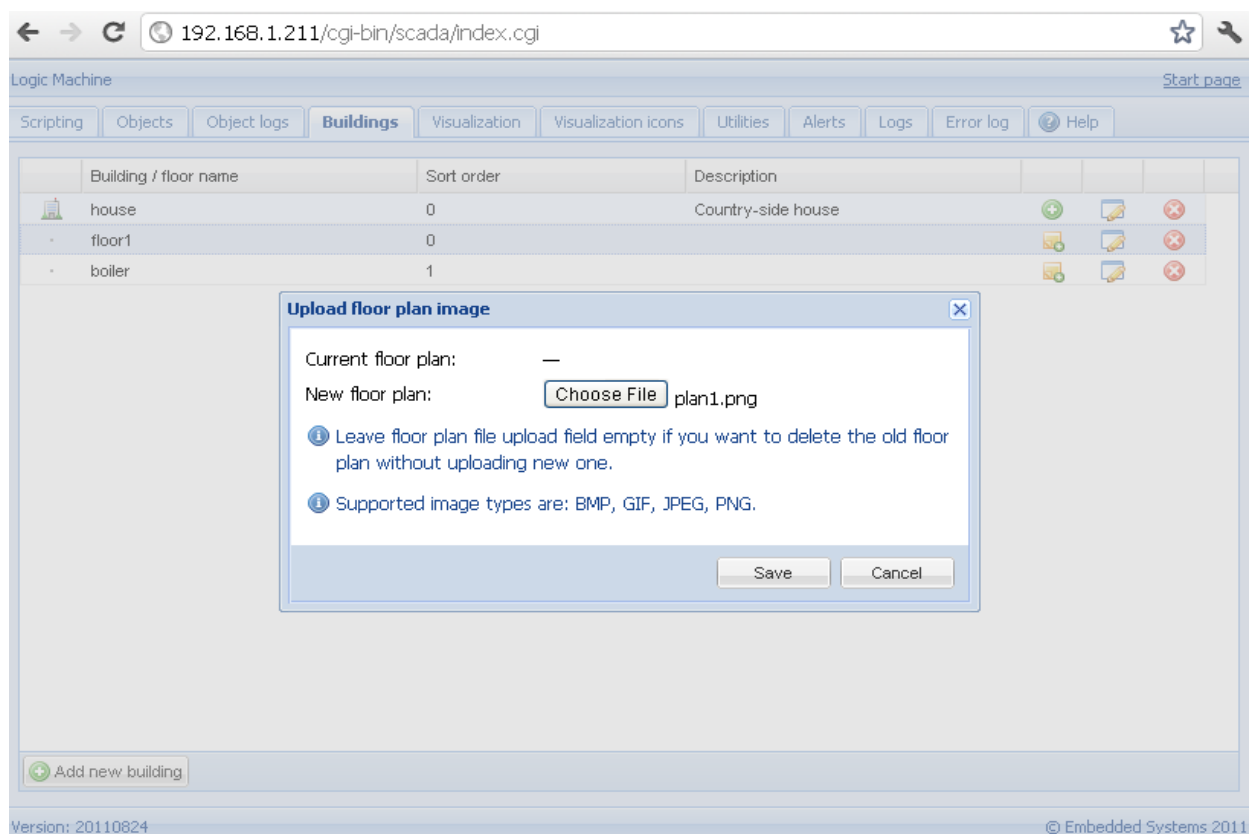




После добавления здания можно определить этажи и загрузить изображения этажей здания. Для добавления этажа нажать на зеленую пиктограмму ; для удаления записи нажать на красную пиктограмму .



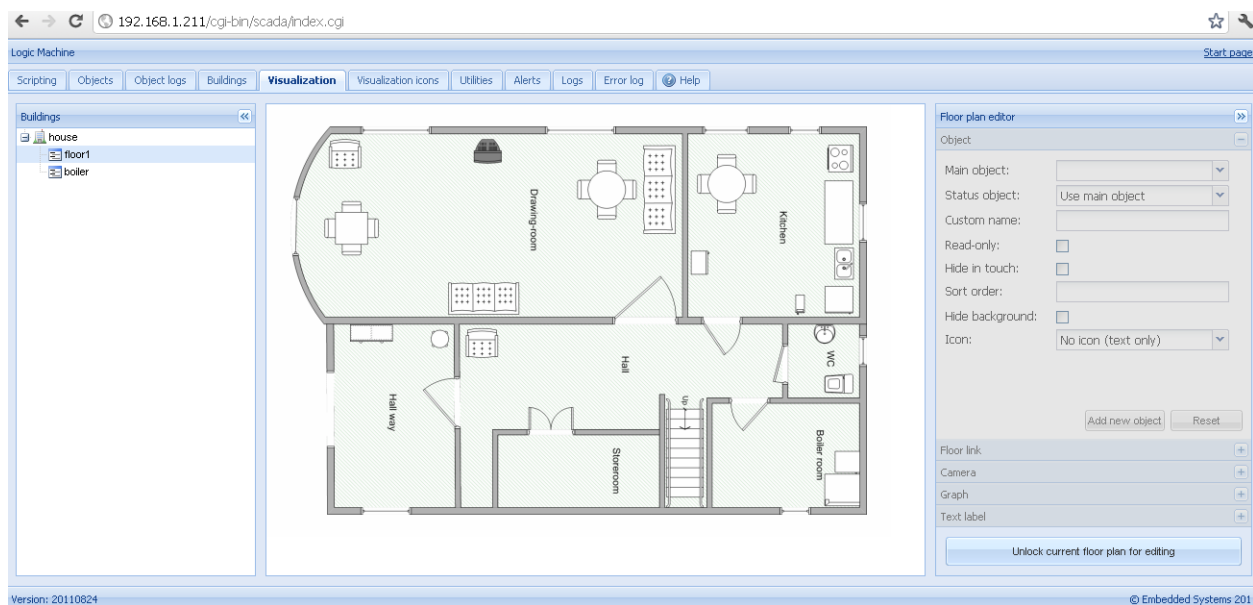
После того как добавили новый этаж загрузите фоновое изображение этажа нажав на кнопку  и выберите файл с локального компьютера. Поддерживаются следующие форматы файлов



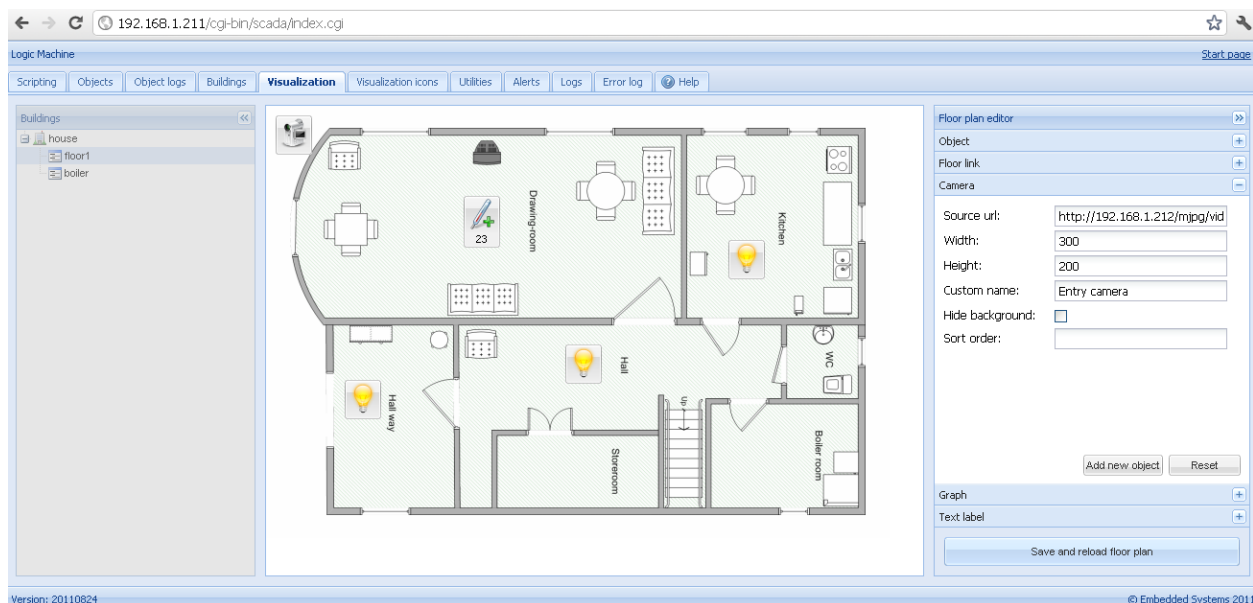
BMP, GIF, JPEG, PNG.

## Добавьте объекты на созданный план этажа

После того как определена структура зданий и этажей во вкладке *Buildings*, она визуализируется во вкладке *Visualization*. Здесь можно добавить объекты на планы этажей. Можно увеличить видимый план этажа минимизировав боковые окна нажатием на пиктограммы правой и левой стрелочки. Это особенно важно для маленьких дисплеев



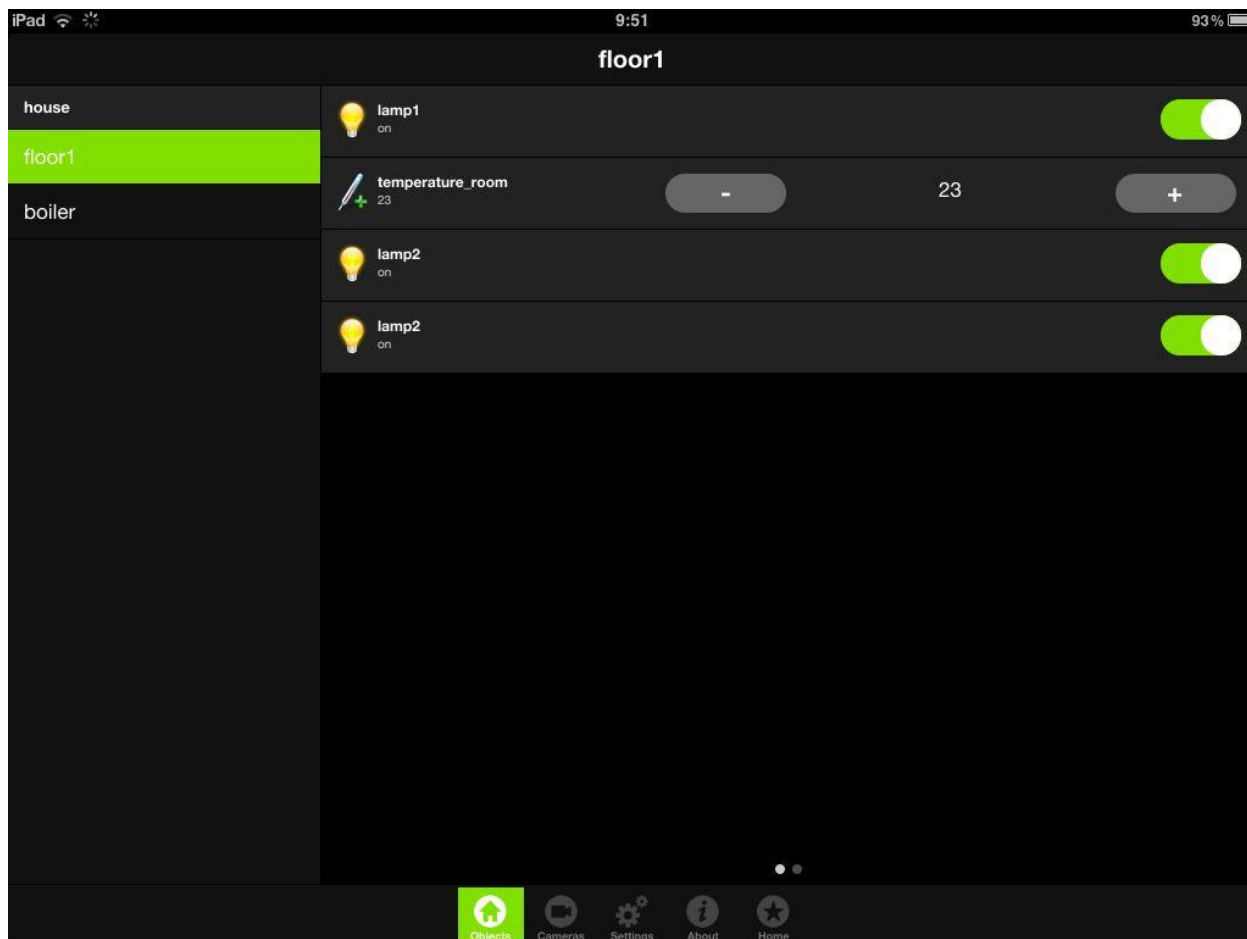
Для добавления объектов на план нужно нажать на кнопку *Unlock current floor plan for editing*. После того как параметры объекта определены нажмите на кнопку *Add new object* для добавления на план этажа. Вы можете двигать объект в любое место плана. Внимание - пока объект находится в режиме редактирования, он еще не работает.



Когда все объекты и камеры будут добавлены нажмите кнопку *Save and reload floor plan* для выхода из режима редактирования.

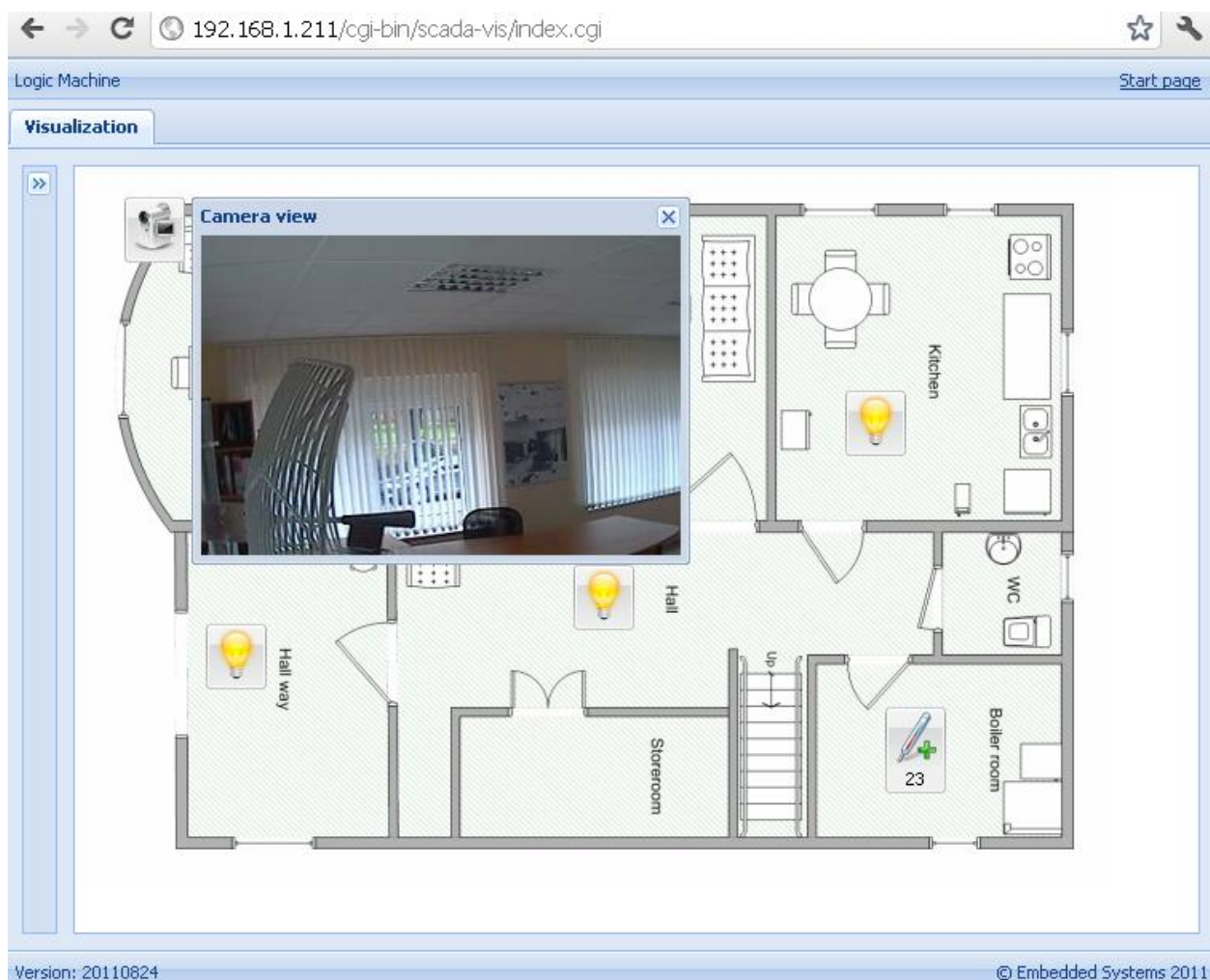
### Настройка визуализации на сенсорном устройстве (iPad в этом случае)

- Проверьте есть ли беспроводное соединение между вашим iPad и Logic Machine (через точку доступа).
- Введите в браузере IP адрес Logic Machine (по умолчанию 192.168.0.10).
- Нажмите на пиктограмму Touch Visualization.
- Сохраните ярлык на домашнем экране вашего iPad



### Настройка визуализации на PC, Mac

- Проверьте есть ли соединение между вашим PC/Mac и Logic Machine и введите в браузере IP адрес Logic Machine (по умолчанию 192.168.0.10)
- Нажмите на пиктограмму *User Mode Visualization* и выберите «этаж» который хотите видеть.
- Для лучшего просмотра плана этажа минимизируйте боковое окно нажав пиктограмму со стрелкой.

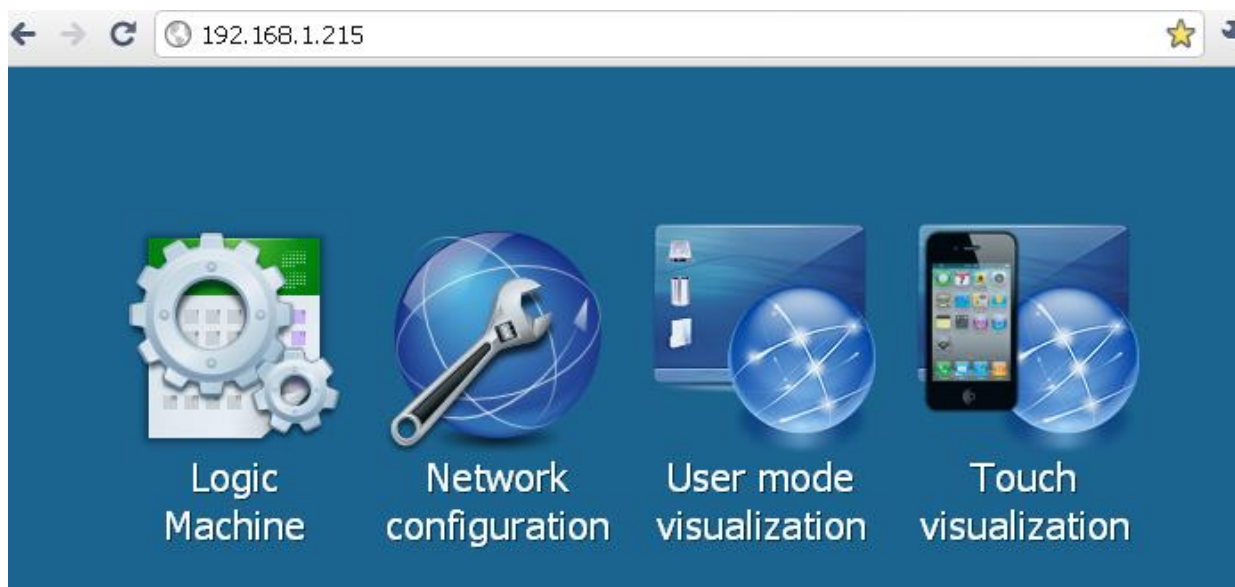


## Вход в графический интерфейс пользователя

По умолчанию для Logic Machine 2 в интерфейсе LAN задан следующий IP-адрес: 192.168.0.10. Ввести вышеупомянутый IP-адрес в адресной строке браузера.

*Внимание! Убедитесь, что используются IP-настройки соединения компьютера и Logic Machine 2 одной подсети.*

После входа выводится страница по умолчанию.



- **Logic Machine 2** – настройка параметров KNX: скрипты, связи между объектами, предупреждения и объекты KNX, визуализация управления зданием.
- **Network configuration** – настройка IP и KNXnet/IP.
- **User mode visualization** – определение отображения объектов управления.
- **Touch visualization** – система отображения данных для сенсорных устройств iPhone/iPod/iPad/Android.

## Окно Logic Machine 2

Имя пользователя	Пароль
admin	admin

Исходный каталог для настройки конфигурации Logic Machine 2. Главное меню включает следующие подменю:

*Scripting* – управление архивом скриптов.

*Objects* – перечень сетевых объектов KNX.

*Object logs*– журнал событий объектов шины KNX.

*Building*– загрузка сведений о здании и файла изображений.

*Visualization*– управление и контроль графического отображения данных.

*Visualization icons*– управление пиктограммами.

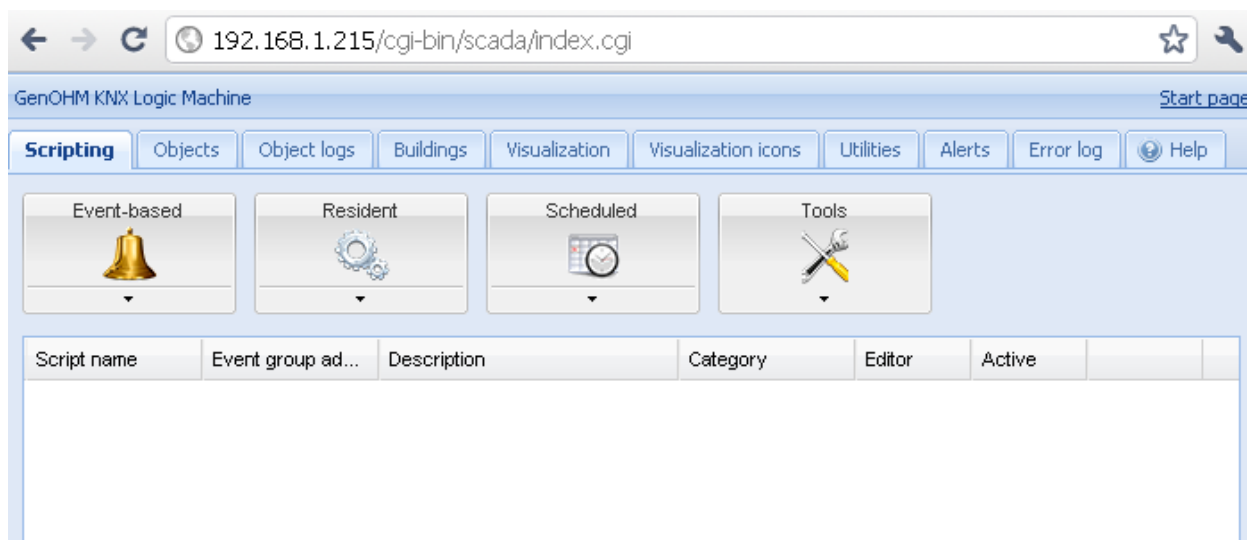
*Utilities*– сервисные программы (в т. ч. импорт из ETS, сброс базы данных объектов, резервное копирование, обновление системы).

*EnOcean* – управление устройствами EnOcean

*Alerts* – предупредительные сообщения, заданные с использованием функции **alert**.

*Error log* – сообщения об ошибках на шине KNX.

*Help* – справка по синтаксису скриптов.



## Меню Scripting

Настоящее меню позволяет добавлять и управлять скриптами различных типов. Для создания пользовательских скриптов используется язык программирования Lua. Основные моменты, связанные с применением данного языка программирования, описаны в первом издании руководства «Программирование на языке Lua», которое находится в свободном доступе по адресу: <http://lua.org/pil/>.



*Внимание! Формат данных: в большинстве случаев данные хранятся и передаются друг другу компонентами Logic Machine 2 с использованием закодированных в шестнадцатеричном формате строк (2 байта /1 байт данных).*

### Три основных вида сценариев:

*Основанные на событиях* – скрипты, выполняемые при наступлении определенного группового события на шине данных. Как правило, применяются при необходимости получения ответа максимально приближенно к реальному режиму времени.

*Резидентные* – скрипты, использующие упорядоченный опрос для проверки статуса объектов. Как правило, применяются в сочетании с системами вентиляции и отопления, когда данные принимаются от нескольких групповых адресов.

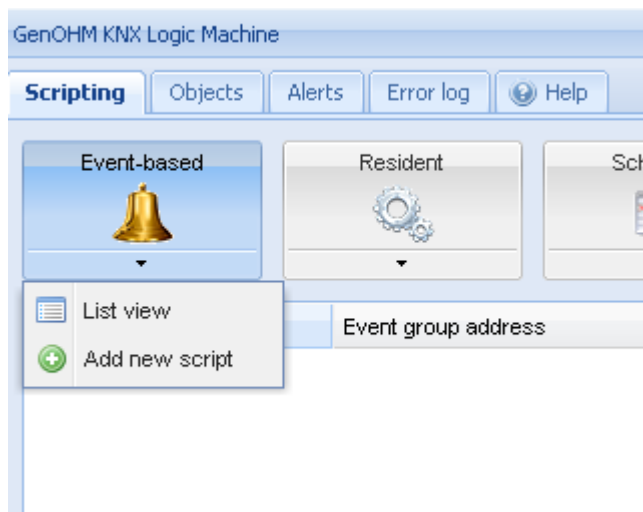
*Запланированные* – скрипты, выполняемые в определенное время. Используются в сочетании с различными системами безопасности и для имитации присутствия.

## Добавление нового скрипта

После нажатия на стрелку в нижней части кнопки Event-based, Resident или Scheduled выводятся два пункта меню:

*List view* – отобразить список скриптов.

*Add new script* – добавить в список новый скрипт.



## Сценарии, основанные на событиях

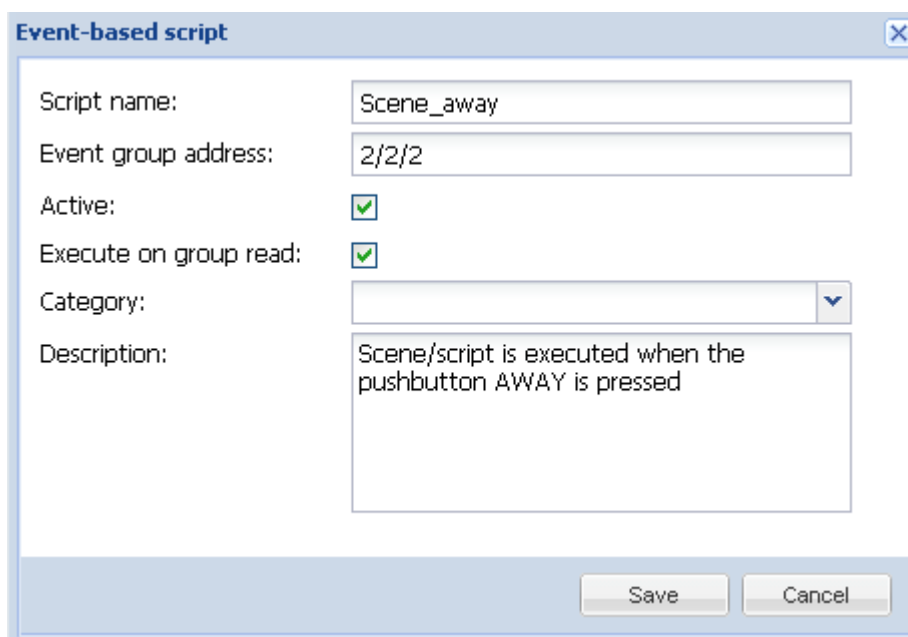
Сценарии, основанные на событиях, могут использоваться для задания логики обработки событий, связанных с групповым адресом. Функция, установленная пользователем, выполняется при наступлении события *group write* для указанного группового адреса. Данные о событии хранятся в глобальной переменной типа **event**. Значения переменной:

- `dstraw (integer)` – закодированный групповой адрес назначения;
- `srcraw (integer)` – закодированный индивидуальный адрес источника;
- `dst (string)` – расшифрованный групповой адрес назначения (пример: 1/1/4);
- `src (string)` – расшифрованный индивидуальный адрес источника (пример: 1.1.2);
- `type (string)` – тип события: `groupwrite`, `groupread` или `groupresponse`. В настоящее время пользовательские скрипты привязаны только к событиям типа `groupwrite` и `groupread`;
- `dataraw (integer/string)` – закодированные двоичные данные;
- `datahex (string)` – данные в виде шестизначной кодированной строки; используются для преобразования значения в переменную Lua.


*Внимание! Переменная **event** доступна только для функций, основанных на событиях, и недоступна для резидентных и запланированных функций.*

*Внимание! Все основанные на событиях скрипты выполняются в рамках единой очереди. Убедитесь, что скрипты, основанные на событиях, не содержат бесконечные циклы, команды на переход в ждущий режим и иные блокирующие команды.*

**Следующие поля должны быть заполнены при добавлении нового скрипта:**




*Script name* – имя скрипта.

*Event group address* – значение группового адреса. Можно использовать только цифровые значения от 0 до 9 и / в качестве разделителя.  в правой части текстового поля означает, что используется неверный адрес. Верный групповой адрес (пример): 1/1/1.


*Description* – описание скрипта.

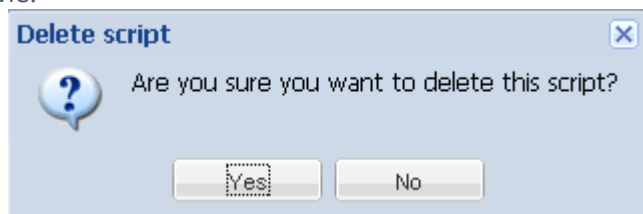
*Category* – имя группы, к которой относится скрипт. Новая или существующая группа скриптов для сохранения нового скрипта. Данный параметр не влияет на выполнение скрипта и помогает сортировать скрипты, а также просматривать их по группам на странице Tools>Print script listings.

*Editor*  – возможность вносить изменения в скрипт с фрагментами кода в редакторе скриптов.

*Active* – включенный (зеленый) / выключенный (красный) скрипт.



 – пиктограмма появляется справа от колонки *Active* после добавления скрипта. При нажатии на пиктограмму скрипт будет удален. При этом выводится окно с требованием подтвердить удаление.

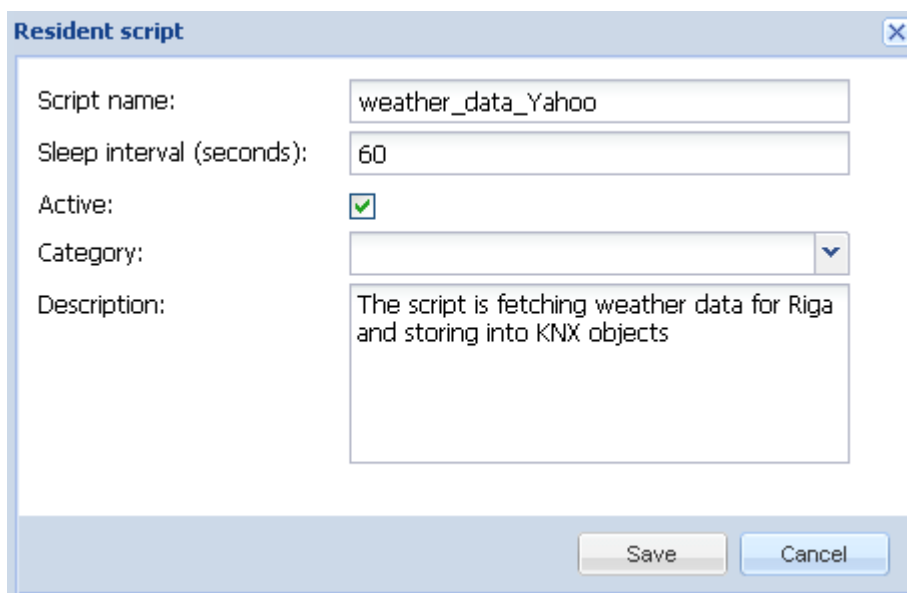


## Резидентные скрипты

Резидентные скрипты выполняются бесконечное число раз. Скрипты выключаются после каждого вызова; выполнение скриптов возобновляется после истечения установленного времени задержки.

*Внимание! Хотя резидентные скрипты выполняются одновременно, они не должны содержать бесконечные циклы; при этом возможность повторной загрузки скриптов после редактирования отсутствует*

**Следующие поля должны быть заполнены при добавлении нового скрипта:**

A dialog box titled "Resident script" with a close button. It contains several fields: "Script name:" with the value "weather\_data\_Yahoo", "Sleep interval (seconds):" with the value "60", "Active:" with a checked checkbox, "Category:" with a dropdown menu, and "Description:" with a text area containing "The script is fetching weather data for Riga and storing into KNX objects". At the bottom are "Save" and "Cancel" buttons.


*Script name* – имя скрипта.


*Script sleep interval*– период времени в секундах, по истечении которого скрипт выполняется повторно.

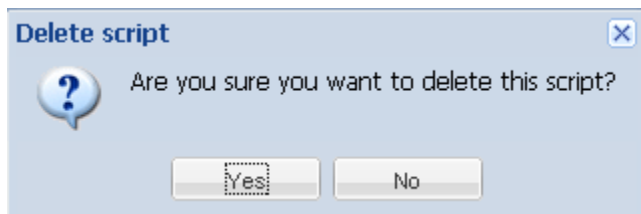
*Active* – включенный (зеленый) / выключенный (красный) скрипт.

*Category* – группа, к которой относится скрипт.

*Description* – описание скрипта.

*Editor*  – возможность вносить изменения в скрипт с фрагментами кода в редакторе скриптов.

 – пиктограмма появляется справа от колонки *Active* после добавления скрипта. При нажатии на пиктограмму скрипт будет удален. При этом выводится окно с требованием подтвердить удаление.



## Запланированные скрипты

Запланированные скрипты выполняются, когда системное время совпадает с заданным временем запуска скрипта. Запланированные скрипты выполняются только раз после каждого срабатывания таймера.

**Следующие поля должны быть заполнены при добавлении нового скрипта:**

A dialog box titled "Scheduled script" with a close button. It contains several fields for scheduling a script: "Script name:" with the value "Floor heating off"; "Minute:" with a question mark icon and the value "0"; "Hour:" with a question mark icon and the value "8,19"; "Day of the month:" with a question mark icon and the value "\*"; "Month of the year:" with a dropdown menu showing "Every month of the year"; "Day of the week:" with a dropdown menu showing "Every day of the week"; "Active:" with an unchecked checkbox; "Category:" with a dropdown menu; and "Description:" with a text area containing "Turns floor heating OFF at 8:00 and 19:00". At the bottom right are "Save" and "Cancel" buttons.

*Script name* – имя скрипта.

Ниже заполняются настройки, когда запускается скрипт

*Minute* – минута

*Hour* – час

*Day of the month* – в какие дни месяца

*Month of the year* – в какие месяцы года

*Day of the week* – в какие дни недели

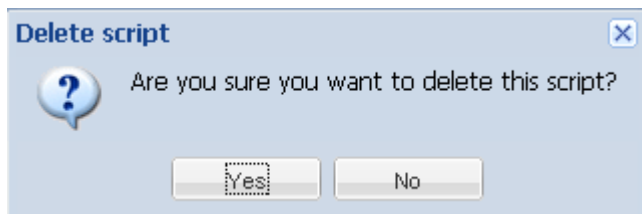
*Active* – включить выполнение скрипта (зеленый) или отключить (красный)


*Category* – группа, к которой относится скрипт.

*Description* – описание скрипта.

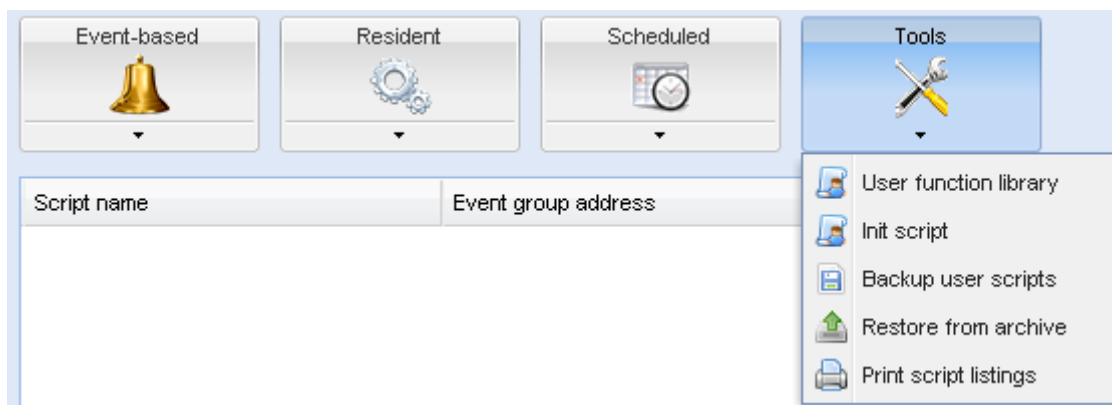


*Editor* – возможность вносить изменения в скрипт с фрагментами кода в редакторе скриптов.



 – пиктограмма появляется справа от колонки Active после добавления скрипта. При нажатии на пиктограмму скрипт будет удален. При этом выводится окно с требованием подтвердить удаление.

## Инструменты

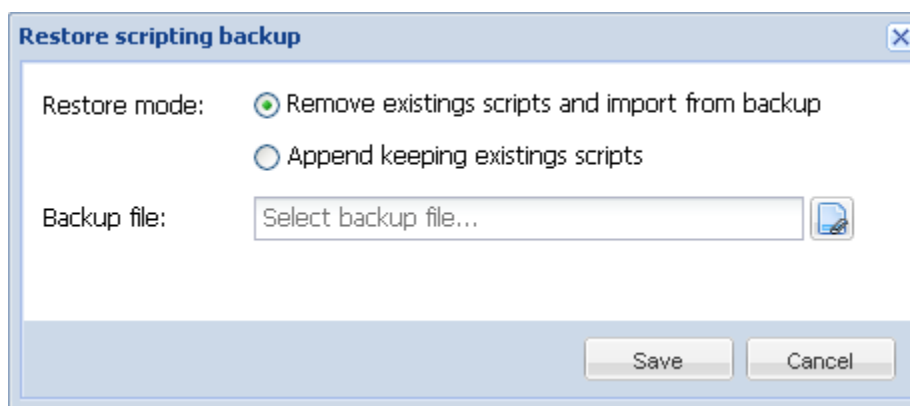


*User function library* – библиотека пользовательских функций.

*Init script* – запустить инициализирующий скрипт.

*Backup user scripts* – резервное копирование всех скриптов в файле \*.gz.

*Restore from archive* – копирование скрипта из архива (\*.gz); доступно две функции:  
удалить все скрипты и импортировать из резервной копии;  
добавить к существующим скриптам (s).



*Print script listings* – отображаются все группы скриптов с кодами (в виде списка).

## Category: Presence

### Presence simulator (id: 1)

Type: Resident

Active: Yes

Script sleep interval: 20


Synchronizes 0/0/2 value with 0/0/1

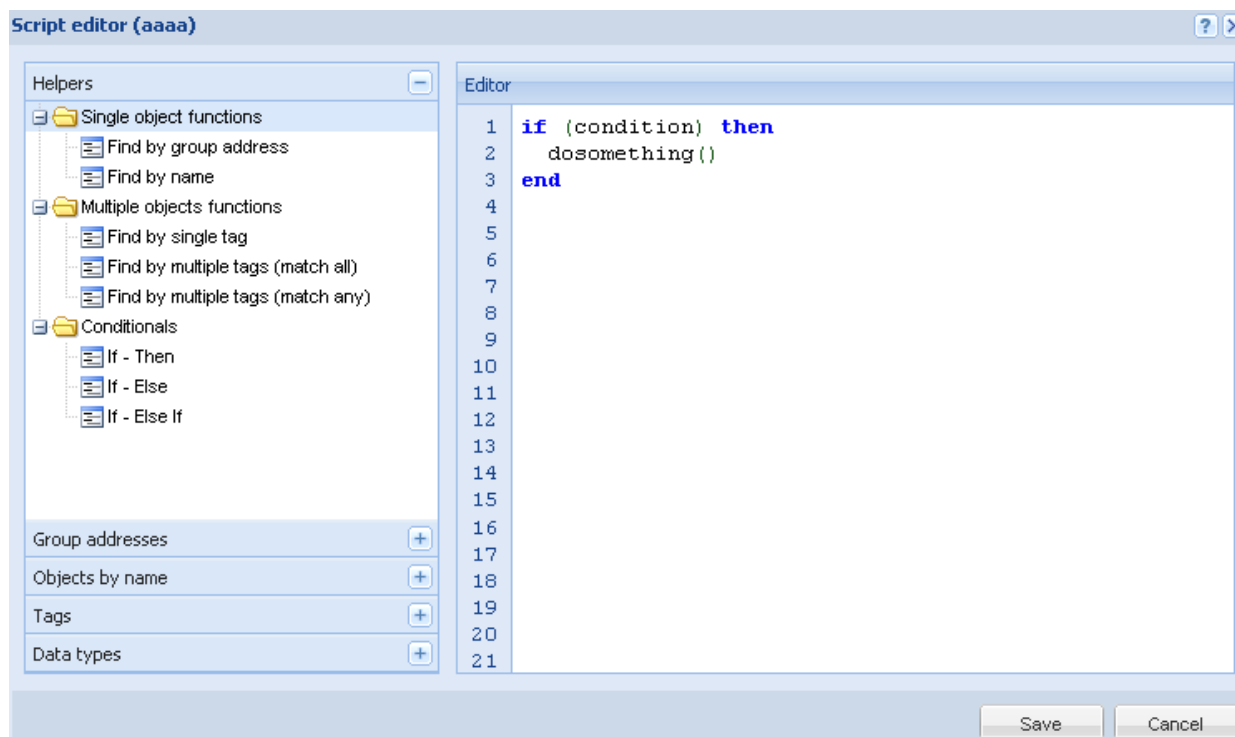
```
-- if object exists "presence" variable will be a table, nil otherwise
presence = knxobject.get('address', '0/0/1')



-- check that object exists and data has been decoded
if presence and presence.decoded then
  -- result will be either "value = true" or value = "false"
  alert('value = %s', tostring(presence.data))

  -- update 0/0/2 with the same data
  knxobject.write('0/0/2', presence.data, dt.bool)
else
  alert('read error')
end
```

## Редактор скриптов. Синтаксис

После добавления скрипта в графе *Editor* появляется пиктограмма ; это означает, что пользователь может открыть скрипт в режиме редактирования и внести необходимые изменения с использованием заранее заданных фрагментов кода.



В случае если пользователь не знаком с синтаксисом, выводится подсказка, где описан порядок создания скриптов. Фрагменты кода экономят время и облегчают процесс программирования. После выбора соответствующего фрагмента кода программа автоматически добавляет его в поле редактирования. В правом верхнем углу отображается ярлык справки и закрытия окна ( ).

### Пять основных разделов редактора скриптов:

- Helpers – заранее заданные фрагменты кода (в частности, функции «если, то»).
- Данная группа включает одиннадцать подгрупп:
- Conditional – условные функции объектов.
  - Loops and iterators – циклы и итераторы (Array, Repeat..Until так далее)
  - Math – математические операции Random, Ceiling, Absolute, Round и так далее.
  - Objects/KNX bus – функции работы с протоколом KNX
  - Storage – работа с локальной базой данной
  - Alerts and logs – Alert, Log variables, Formatted alert
  - Time functions – функция задержки выполнения скриптов
  - Miscellaneous – функции расчета времени заката и рассвета.
  - Modbus – функции работы с протоколом Modbus.

- Serial – функции работы со встроенными сериальными портами ввода-вывода Logic Machine
- Internal IO – функции работы со встроенными портами ввода-вывода (аналоговые и бинарные входы, бинарные выходы)
- DMX – функции работы с протоколом DMX

- *Group addresses* – существующие групповые адреса шины KNX.
- *Objects by name* – выбор объекта по имени.
- *Tags* – выбор объекта по тегу.
- *Data types* – выбор объекта по типу данных.

## Функции объектов

*grp* объект обеспечивает удобный доступ к объектам системы по групповому адресу.

Большинство функций используют *alias* – это может быть групповой адрес или символьное имя (например '1/1/1' или 'My object'). Символьное имя задается при описании объекта

### *grp.getvalue(alias)*

Возвращает значение объекта or Lua *nil* если объект не найден.

### *grp.find(alias)*

Возвращает объект по заданному *alias*. Значение объекта может быть декодировано автоматически только если тип данных был определен в модуле Objects. Если объект не найден то функция вернет Lua *nil*, иначе функция вернет Lua *table* со следующими полями:

- *address* — групповой адрес объекта
- *updatetime* — время последнего обновления в формате UNIX timestamp. Используйте функцию Lua *os.date()* для конвертирования даты в формат для чтения

Если тип данных определен в модуле 'Objects' то будут доступны следующие поля:

- *name* — уникальное имя объекта
- *datatype* — тип данных объекта, определенный пользователем
- *decoded* — имеет значение *true* когда декодированное значение доступно
- *value* — декодированное значение объекта

### *grp.tag(tags, mode)*

Возвращает Lua *table* содержащий объекты имеющих определенную метку. Параметр Tags может быть как Lua *table* или строка. Параметр Mode может быть 'all' (возвращает объекты, которые имеют все данные метки) или 'any' (по умолчанию — возвращает объекты, которые имеют хотя бы одну из данных меток). Вы можете использовать *Returned object functions* для работы над полученной *table*.

### *grp.alias(alias)*

Конвертирует групповой адрес в имя объекта или имя в адрес. Возвращает Lua *nil* когда объект не может быть найден.

## Функции, связанные с объектом

Объекты, полученные с использованием *grp.find(alias)* или *grp.tag(tags, mode)*, имеют приведенные ниже функции.

*Внимание! В обязательном порядке убедитесь в том, что искомый объект был найден. В противном случае обращение к данным функциям приведет к ошибке. Пример приведен ниже.*

### *object:write (data, datatype)*

Групповой запрос на запись данных по указанному групповому адресу объекта. Параметр *datatype* опционален и требуется только в том случае, если тип данных не определен в модуле Objects. Функция возвращает переменную типа `Lua boolean` как результат работы

### *object:response(data, datatype)*

См. *object:write*. Запрос на ответ по указанному групповому адресу.

### *object:read()*

Запрос на считывание данных по указанному групповому адресу.

### *object:update(value, datatype)*

Функция подобна *object:write*, но не посылает новое значение на шину. Применяется для объектов, которые используются только в визуализации.

## Функции для работы с групповыми адресами

### *grp.write(alias, value, datatype)*

Записывает значение по *alias*. Тип данных берется из модуля Objects, если не определен третий параметр. Функция возвращает переменную типа `Lua boolean` как результат работы.

### *grp.response(alias, value, datatype)*

См. *grp.write*. Запрос на ответ по указанному групповому адресу..

### *grp.read(alias)*

Запрос на считывание данных по указанному групповому адресу.

### *grp.update(alias, value, datatype)*

Функция подобна *grp.write*, но не посылает новое значение на шину. Применяется для объектов, которые используются только в визуализации.



## Примеры использования скриптов

### Найти объект по имени и записать новое значение.

```
1.myobject = grp.find('My object')
2.-- grp.find вернет nil если объект не был найден
3.if myobject then
4.  myobject:write(1) - устанавливаем значение объекта 1
5.end
```

### Найти объект по адресу и записать новое значение.

```
1.myobject = grp.find('1/1/15')
2.- проверяем что запрошенный объект был найден
3.if myobject then
4.  myobject:write(52.12, dt.float16) - устанавливаем тип данных как dt.float16 (2-байтное
    число с плавающей точкой)
5.end
```

### Выключаем все объекты с меткой 'lights'.

```
1.lights = grp.tag('lights')
2.lights:write(false)
```

### Запись значений определенного типа данных в групповые адреса.

```
1.grp.write('1/1/1', true, dt.bool) -- пишем 1-бит 'on' в 1/1/1
2.grp.write('1/1/2', 50, dt.scale) -- пишем 1-байт 50% в 1/1/2
```

## Функции типов данных, типы данных

Объект *knxdatatype* обеспечивает кодирование и расшифровку данных в формате Lua и KNX.

### *knxdatatype.decode (value, datatype)*

Преобразование шестнадцатеричных кодированных данных в переменные Lua исходя из указанного типа данных. Тип данных указывается как основной (целое число от 1 до 16) или вспомогательный тип данных KNX (целое число от 1000 до 16000).

Возвращаемые значения:

- успешное выполнение – расшифрованные данные в формате Lua (тип зависит от типа данных), длина значения в байтах;
- ошибка – ноль, строка ошибки.

## Типы данных

Для кодирования и расшифровки данных KNX могут использоваться приведенные ниже типы данных. Ниже указаны данные на языке Lua и предварительно заданные константы (жирным шрифтом).

- 1. 1 bit (boolean) –**dt.bool**–логический.
- 2. 2 bit (1 bit controlled) –**dt.bit2** – числовой.
- 3. 4 bit (3 bit controlled) –**dt.bit4** – числовой.
- 4. 1 byte ASCII character –**dt.char** – строковый.
- 5. 1 byte unsigned integer –**dt.uint8**–числовой.
- 6. 1 byte signed integer –**dt.int8** – числовой.
- 7. 2 byte unsigned integer –**dt.uint16** – числовой.
- 8. 2 byte signed integer –**dt.int16** – числовой.
- 9. 2 byte floating point –**dt.float16** – числовой.
- 10. 3 bytetime/day–**dt.time** – таблица со следующими данными:
  - day – числовое значение (0–7);
  - hour – числовое значение (0–23);
  - minute – числовое значение (0–59);
  - second – числовое значение (0–59).
- 11. 3 bytedate–**dt.date** – таблица со следующими данными:
  - day – числовое значение (1–31);
  - month – числовое значение (1–12);
  - year – числовое значение (1990–2089).
- 12. 4 byte unsigned integer –**dt.uint32** – числовой.
- 13. 4 byte signed integer –**dt.int32** – числовой.
- 14. 4 byte floating point –**dt.float32** – числовой.
- 15. 4 byte access control–**dt.access** – числовой (в настоящее время поддерживается не в полном объеме).
- 16. 14 byte ASCII string–**dt.string** – текстовый, нулевые символы ('\0') при расшифровке не учитываются.

## Общие функции. Хранение данных. Предупредительные сообщения

### Хранение данных

Объект *storage* обеспечивает долгосрочное хранение основных данных для скриптов пользователя. Поддерживаются исключительно приведенные ниже типы данных Lua:

- логические,
- числовые,
- строковые,
- табличные.

***storage.set (key, value)***

Задание нового значения для выбранного ключа. Предыдущее значение перезаписывается. Возвращается результат операции boolean и в случае false строка с описанием ошибки.

***storage.get (key, default)***

Получение значения ключу; если не найдено значение, то возвращается значение по умолчанию – `default` (`nil`, если значение не указано).

**Примечание:** все пользовательские скрипты хранят данные в одном хранилище. Убедиться, что один и тот же ключ не используется для сохранения различных типов данных.

## Примеры

На приведенных ниже примерах показан базовый синтаксис для `storage.set`. Логическая константа `true` возвращается в том случае, если все параметры заданы верно:

```
result = storage.set('my_stored_value_1', 12.21)
```

В этом примере будет возвращен `false`, потому что мы пытаемся сохранить функцию.

```
testfn = function(t)
    return t * t
end
```

```
result = storage.set('my_stored_value_2', testfn) – ошибка
```

На приведенных ниже примерах показан базовый синтаксис для `storage.get`. Если значение ключа не обнаружено, после первого запроса возвращается `nil`, после второго – `0` (задано по умолчанию).

```
result = storage.get('my_stored_value_3') -- nil если значение не обнаружено
```

```
result = storage.get('my_stored_value_3', 0) – 0, если значение не обнаружено
```

При сохранении таблиц проверить тип возвращаемых данных. Пример: мы создаем сохраняемый элемент с ключом `test_object_data`:

```
objectdata = {}
objectdata.temperature = 23.1
objectdata.scene = 'default'
```

```
result = storage.set('test_object_data', objectdata) – сохранить переменную objectdata
                                     как 'test_object_data'
```

Теперь мы выбираем данные из хранилища. Проверяем тип данных на корректность.

```
objectdata = storage.get('test_object_data')

if type(objectdata) == 'table' then
    if objectdata.temperature > 24 then
        -- принять меры при повышенной температуре
    end
end
```

## Функция формирования предупредительных сообщений

*Alert (message, [var1, [var2, [var3]]])*

Сохранение предупредительного сообщения и текущего системного времени в основной базе данных. Все предупредительные сообщения хранятся в модуле Alerts. Параметры настоящей функции идентичны Lua *string.format*.

### Пример:

```
1.temperature = 25.3
2.if temperature > 24 then
3.  -- отображаемое сообщение: 'Temperature levels are too high: 25.3'
4.  alert('Temperature level is too high: %.1f', temperature)
5.end
```

## Функция журнала

*log(var1, [var2, [var3, ...]])*

Сохраняемые переменные конвертируются в удобный для восприятия вид и сохраняются в базе данных. Их можно просмотреть в модуле "Logs".

### Пример:

```
1.-- Функция log принимает типы данных Lua nil, boolean, число и таблицу (до 5 уровней
   вложенности)
2.a = { key1 = 'value1', key2 = 2 }
3.b = 'test'
4.c = 123.45
5.-- запись всех переданных переменных
6.log(a, b, c)
```

## Формат запланированного времени выполнения скрипта

Для планирования выполнения скриптов используется стандартный (*cron*) формат ввода даты и времени. Допустимые значения:

- \* – выполнять скрипт каждую минуту, час или день;
- \*/N – выполнять скрипт через каждые N минут, часов и дней. N – целое число; скрипт выполняется, если в результате деления текущего значения на N в остатке 0. Например, скрипт, где для часов задано \*/8, будет выполняться при текущем значении 0, 8 и 16;
- N – выполнять скрипт в точно указанную минуту, час и день N;
- N–K – выполнять скрипт, если значение минут, часа и дня от N до K (включительно);
- N, K – можно указать несколько параметров N и N–K (через запятую). Например, если в скрипты задано 15, 50–52 минуты, он выполняется при текущем значении 15, 50, 51 и 52.

# Расширенная библиотека функций

---

## *toboolean(value)*

Конвертирует значение в тип `boolean` по следующим правилам: `nil`, `boolean false`, `0`, пустая строка, строка со значением `'0'` воспринимаются как `false`, все остальное как `true`

## *os.microtime()*

Возвращает два значения: текущее время в секундах и текущее время в наносекундах.

## *os.sleep(delay)*

Остановка выполнения скрипта на `delay` секунд.

## *string.trim(str)*

Возвращает переданную строку `str` без пробелов в начале и конце строки.

## *string.split(str, sep)*

Разбивает переданную строку `str` на подстроки ориентируясь на символ разделителя `sep`. Возвращает результат в виде Lua table.

## Битовые операции

*bit.bnot(x)* – логический «нет»

*bit.band(x1 [, x2...])* - логический «и»

*bit.bor(x1 [, x2...])* - логический «или»

*bit.bxor(x1 [, x2...])* - логический «хор»

*bit.lshift(x, n)* – сдвиг влево на n позиций

*bit.rshift(x, n)* - сдвиг вправо на n позиций

## *knxlib.decodeia(indaddressa, indaddressb)*

Преобразование индивидуального адреса в двоичном формате в строку Lua. Данная функция принимает один или два параметра (интерпретируются как два отдельных байта).

## *knxlib.decodega(groupaddressa, groupaddressb)*

Преобразование группового адреса в двоичном формате в строку Lua. Данная функция принимает один или два параметра (интерпретируются как два отдельных байта).

## *knxlib.encodega(groupaddress, separate)*

Преобразование строки Lua в двоично закодированный групповой адрес. Групповой адрес возвращается как одно значение Lua, когда второй параметр – *false* или *nil*; в противном случае выводятся два отдельных байта.

## *lmcore.hextoint(hexvalue, bytes)*

Преобразование строки с записанным значением в шестнадцатеричном формате в целое число установленной длины в байтах.

***lmcore.inttohex(intvalue, bytes)***

Преобразование целого числа в строку шестнадцатеричном формате установленной длины в байтах.

***lmcore.strtohex(str)***

Преобразование значения записанного в строку в строку со значением в шестнадцатеричном формате.

***lmcore.hextostr(hexstr)***

Преобразование значения записанного в строку в шестнадцатеричном формате в строку

***lmcore.tonumber(value)***

Преобразование переменной в числовой тип по следующим правилам: числа и строки с записанным в них числом остаются как есть, boolean **true** как 1, boolean **false** как 0, все остальное как **nil**

## Объекты

Перечень объектов сети KNX отображается в меню *Objects*. Объекты попадают в перечень в следующих случаях:

- после перехвата телеграмм с шины от неизвестных групповых адресов. В случае перехвата телеграммы объект отображается с неопределенным типом данных и может использоваться при визуализации.
- Добавляется в ручную
- Импортируется из ESF файла

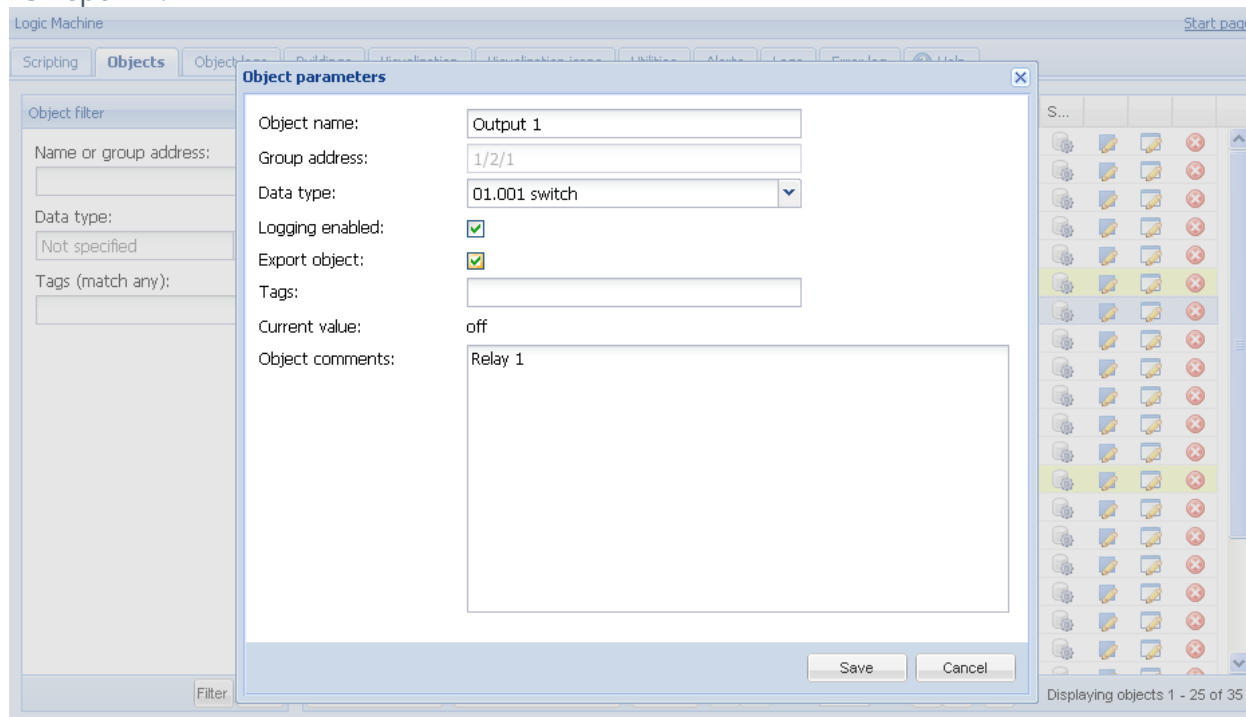
The screenshot shows the 'Logic Machine' software interface. The 'Objects' menu is active, displaying a list of KNX objects. The sidebar on the left contains filters for 'Name or group address', 'Data type' (set to 'Not specified'), and 'Tags (match any)'. The main table lists objects with columns: Group address, Object name, Data type, Current value, Logging enabled, Tags, Object comments, and Set value. The table shows 25 objects, with the first 10 highlighted in yellow. The status bar at the bottom indicates 'Displaying objects 1 - 25 of 35'.

Group a...	Object name	Data type	Current va...	Logging e...	Tags	Object comments	Set value
1/1/1	Button 1	01.001 switch	off	Yes			
1/1/2	Button 2	01.001 switch	off	No			
1/1/3	Button 3	01.001 switch	off	No			
1/1/4	Button 4	01.001 switch	off	No			
1/1/5	Sensor 1	01.001 switch	on	No			
1/1/6		05. 1 byte unsign...	127	No			
1/2/1	Output 1	01.001 switch	off	Yes		Relay 1	
1/2/2	Output 2	01.001 switch	off	No		Relay 2	
1/2/3	Output 3	01.001 switch	off	No		Relay 3	
1/3/1	Status 1	01.001 switch	off	No		Relay 1	
1/3/2	Status 2	01.001 switch	off	No		Relay 2	
1/3/3	Status 3	01.001 switch	off	No		Relay 3	
1/3/4		05. 1 byte unsign...	130	No			
1/4/1	LED On/Off 1	01.001 switch	off	No			
1/4/2	LED On/Off 2	01.001 switch	off	No			
1/4/3	LED On/Off 3	01.001 switch	off	No			
1/4/4	LED Dimmer 1	05.001 scale	0%	No			
1/4/5	LED Dimmer 2	05.001 scale	0%	No			



## Параметры объектов

Для изменения настроек существующих и новых объектов выбрать соответствующий объект из перечня.



*Object name* – имя объекта.

*Group address* – групповой адрес объекта

*Data type* – тип данных KNX объекта. Задается после распознавания Logic Machine 2 нового объекта и подтверждения его параметров.

*Enable logging* – регистрация событий для объекта. Журналы отображаются в меню *Objects logs*.

*Export object* – Делает объект видимым для удаленных XML запросов и в BACnet сети

*Tags* – присвоить метку объекту для последующего создания скриптов (пример: All\_lights\_first\_floor).


*Current value* – текущее значение объекта.

*Object comment* – примечания для объекта.

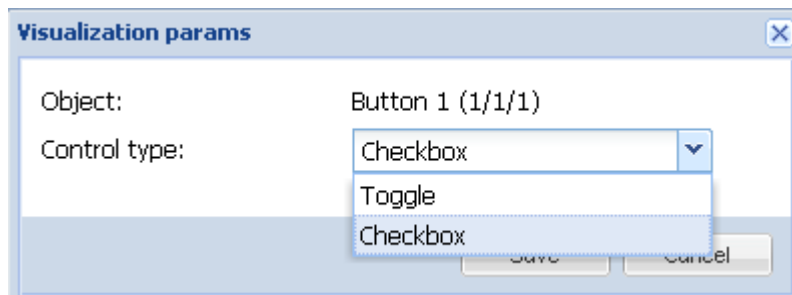
Вы можете сортировать объекты по одному из следующих параметров:

- Имя или групповой адрес объекта
- Тип данных
- Метки

## Параметры визуализации объекта

При нажатии кнопки  выбранного объекта вы можете установить параметры отображения объекта.

### 1 бит



- **Control type** – тип управляющего элемента

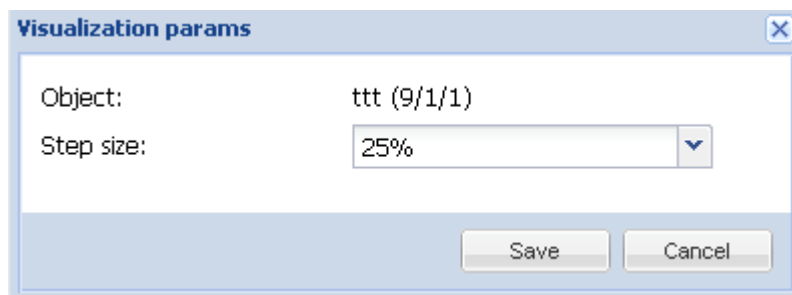
- Переключение



- Чекбокс

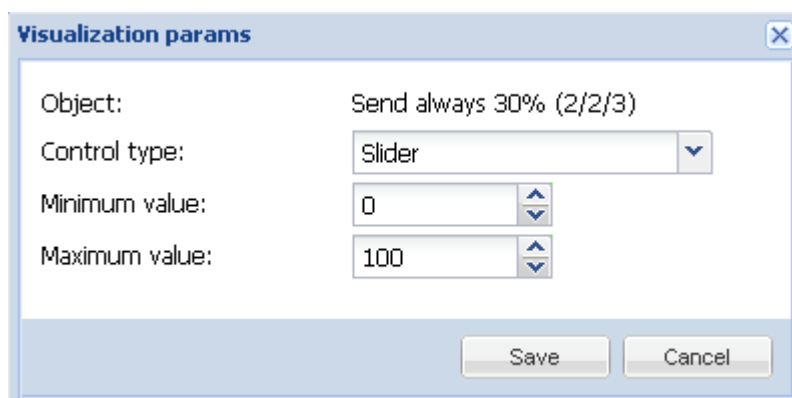


### 4 бит (3 бит управляющие)



- **Step size** – шаг изменения значения (например при управление жалюзи), также это работает со следующими типами объекта:

2 bit (1 bit controlled), 1 byte unsigned integer (scale), 1 byte signed integer, 2 byte unsigned integer, 2 byte signed integer, 2 byte floating point (temperature), 4 byte unsigned integer, 4 byte signed integer, 4 byte floating point



- Control type – тип управляющего элемента
  - Слайдер




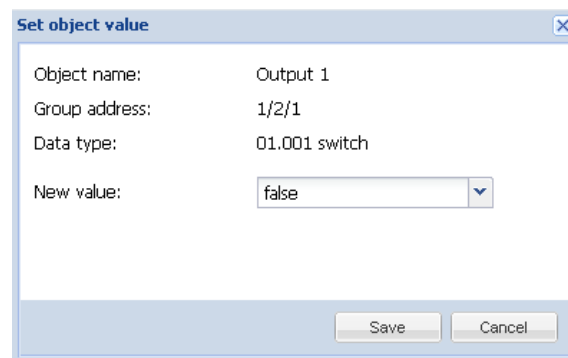
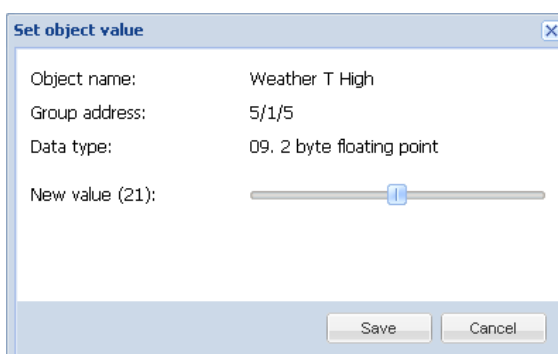
- Прямой ввод / Шаг +/-



- Minimum value – минимальное значение
- Maximum value – максимальное значение

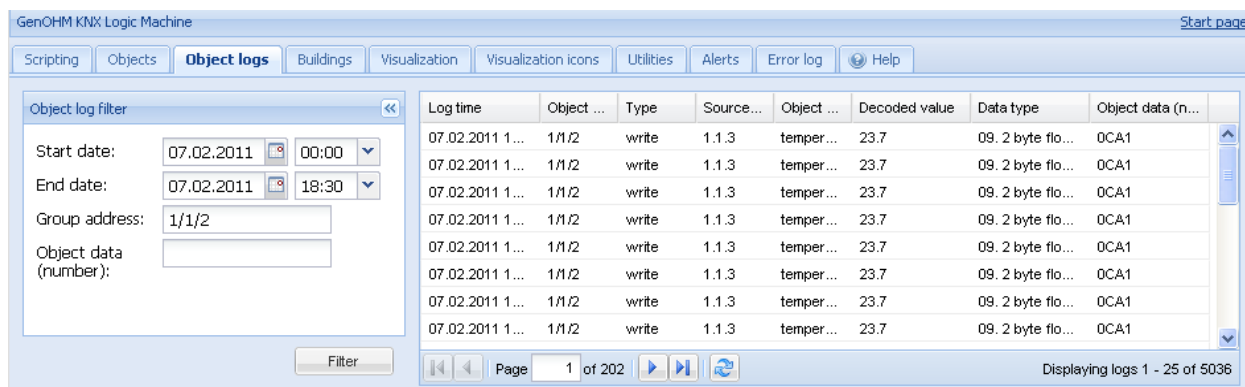
## Изменение состояния объекта

В списке объектов нажав  кнопку, вы можете изменить состояние объекта. В поле *New value* в зависимости от типа параметров визуализации вы устанавливаете новое значение.



## Журнал событий

История телеграмм объектов отображается в меню *Object logs*. После включения данной функции для объекта, в журнале регистрируются все события связанные с этим объектом.



При необходимости поиска информации за определенный период времени задается фильтр.

*Start date* – дата и время начала фильтрации событий в журнале.

*End date* – дата и время окончания фильтрации событий в журнале.

*Group address* – групповой адрес.

*Object data (number)* – данные объекта.

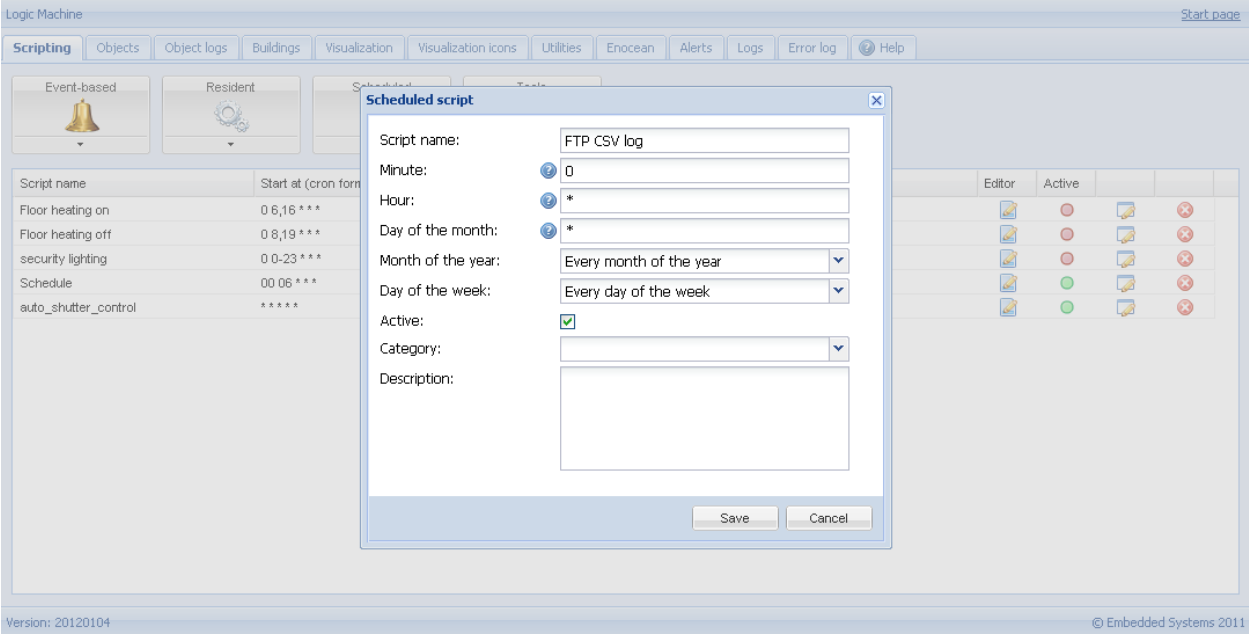
Для очистки **всех** телеграмм от **всех** объектов в журнале, нажмите кнопку Clear

## Экспорт журнала

Например

Каждый час, формируется CSV файл, хранящий все прошедшие за это время события, и посылается на внешний FTP с IP адресом 192.168.1.11, логином 'ftlogin', паролем 'ftppassword'.

- Создаем скрипт по расписанию который вызывается каждый час



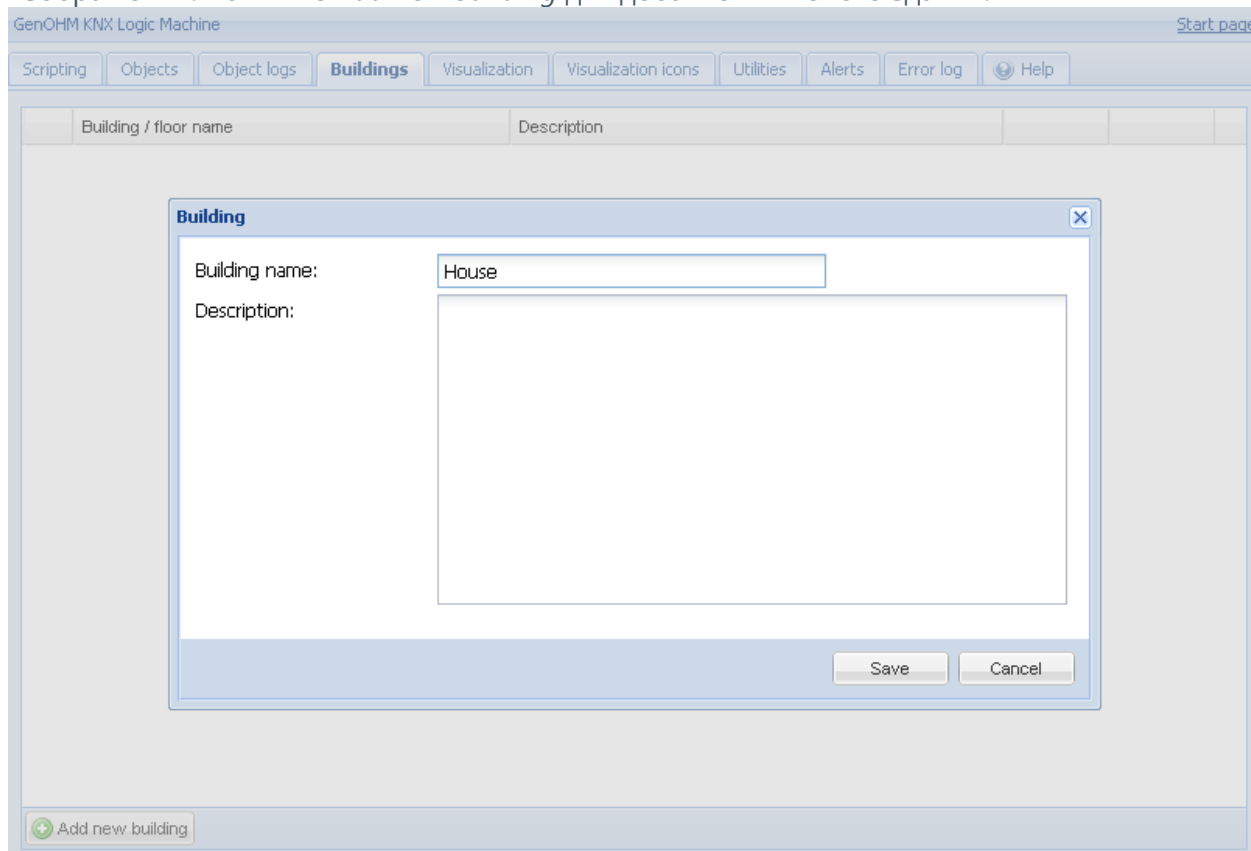
- И добавляем следующий код в редакторе скриптов

```
1. require('socket.ftp')
2.
3. -- ftp файл
4. ftpfile = string.format('ftp://ftplogin:ftppassword@192.168.1.11/%s.csv', os.date('%Y-%m-%d_%H-%M'))
5. -- получаем время час назад (3600 сек)
6. logtime = os.time() - 60 * 60
7.
8. -- Список объектов по id
9. objects = {}
10.
11. -- получаем список объектов с включенным логированием
12. query = 'SELECT address, datatype, name FROM objects WHERE disablelog=0'
13. for _, object in ipairs(db:getall(query)) do
14.     objects[ tonumber(object.address) ] = {
15.         datatype = tonumber(object.datatype),
16.         name = tostring(object.name or ''),
17.     }
18. end
19.
20. -- csv буфер
21. buffer = { '"date","address","name","value"' }
22.
23. -- получаем список событий за последний час
24. query = 'SELECT src, address, datahex, logtime, eventtype FROM objectlog WHERE logtime >=
? ORDER BY id DESC'
25. for _, row in ipairs(db:getall(query, logtime)) do
26.     object = objects[ tonumber(row.address) ]
27.
28.     -- находим соответствие между объектами и событиями типа «write»
29.     if object and row.eventtype == 'write' then
30.         datatype = object.datatype
31.
32.         -- проверяем тип объекта
33.         if datatype then
34.             -- декодируем данные
35.             data = knxdatatype.decode(row.datahex, datatype)
36.
37.             -- удаляем пробелы и служебные символы если это строка
38.             if datatype == dt.char or datatype == dt.string then
39.                 data = data:gsub('%z+', '')
40.             -- дату конвертируем в формат DD.MM.YYYY
41.             elseif datatype == dt.date then
42.                 data = string.format('%02d.%02d.%02d', data.day, data.month, data.year)
43.             -- время в HH:MM:SS
44.             elseif datatype == dt.time then
45.                 data = string.format('%02d:%02d:%02d', data.hour, data.minute, data.second)
```



```
46.     end
47.     else
48.         data = ''
49.     end
50.
51.     -- форматируем в строку формата csv
52.     logdate = os.date('%Y.%m.%d %H:%M:%S', row.logtime)
53.     csv = string.format('%q,%q,%q,%q', logdate, knxlib.decodega(row.address),
        object.name, tostring(data))
54.
55.     -- добавляем в буфер
56.     table.insert(buffer, csv)
57. end
58. end
59.
60. -- загружаем на ftp сервер только если есть данные в буфере
61. if #buffer > 1 then
62.     result, err = socket.ftp.put(ftpfile, table.concat(buffer, '\r\n'))
63. end
64.
65. -- обработка ошибок при загрузке
66. if err then
67.     alert('FTP upload failed: %s', err)
68. end
```

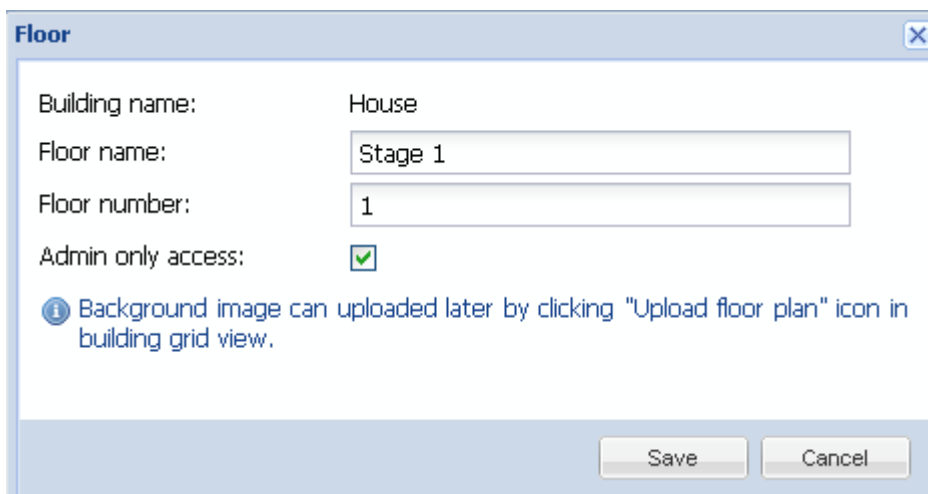
## Здания

Во вкладке Building задается структура здания, определяются этажи, загружаются фоновые изображения. Нажмите *Add new building* для добавления нового здания.





После добавления здания можно определить этажи и загрузить изображения этажей здания. Для добавления этажа нажать на зеленую пиктограмму ; для удаления записи нажать на красную пиктограмму .




**Floor**

Building name: House

Floor name: Stage 1

Floor number: 1

Admin only access: ☒

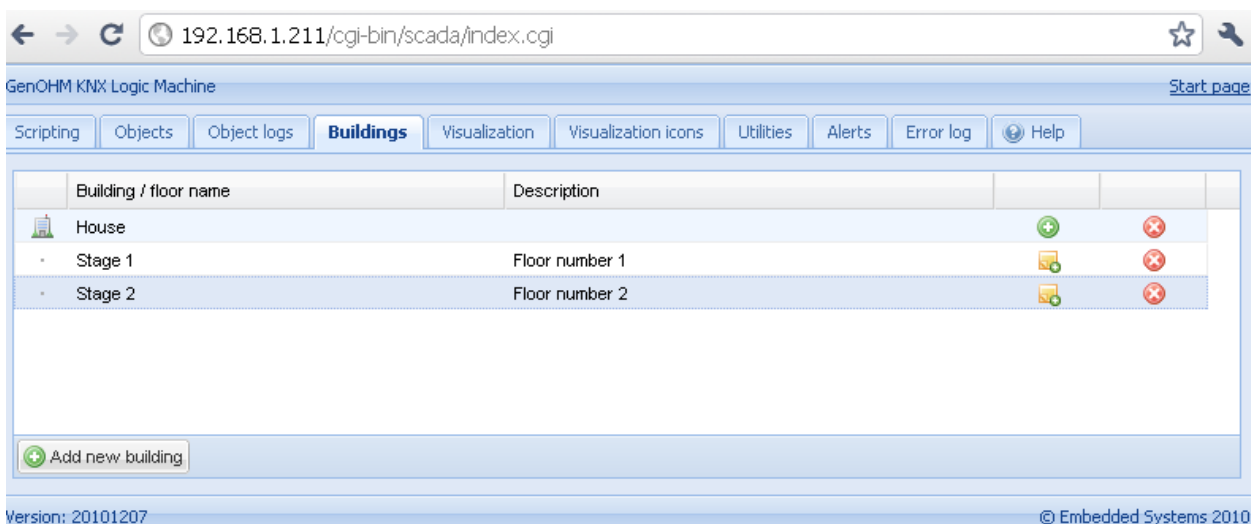
 Background image can uploaded later by clicking "Upload floor plan" icon in building grid view.


Save Cancel

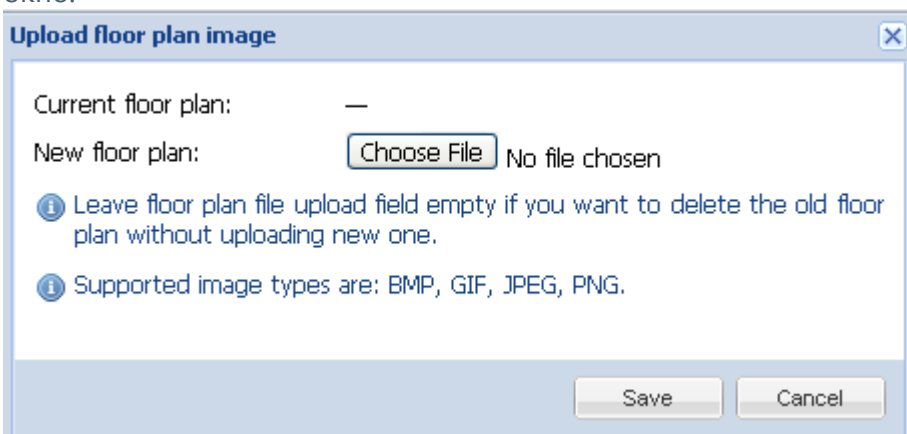
*Floor name* – название этажа.

*Floor number* – номер этажа.

*Admin access only* – установить допуск к данным этажа только для пользователей с правами администратора.




Нажать на , чтобы добавить новое фоновое изображение для этажа; выводится следующее окно:




**Upload floor plan image**


Current floor plan: —

New floor plan:  No file chosen

 Leave floor plan file upload field empty if you want to delete the old floor plan without uploading new one.

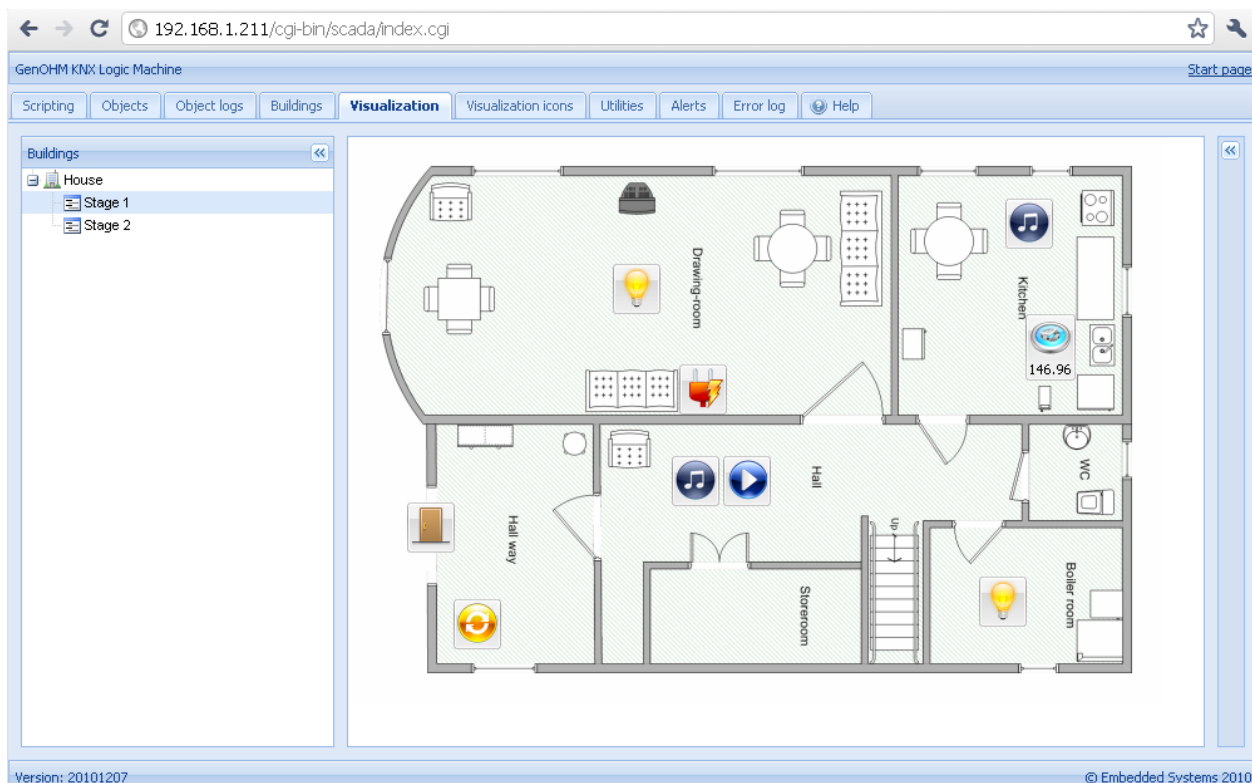
 Supported image types are: BMP, GIF, JPEG, PNG.


Save Cancel

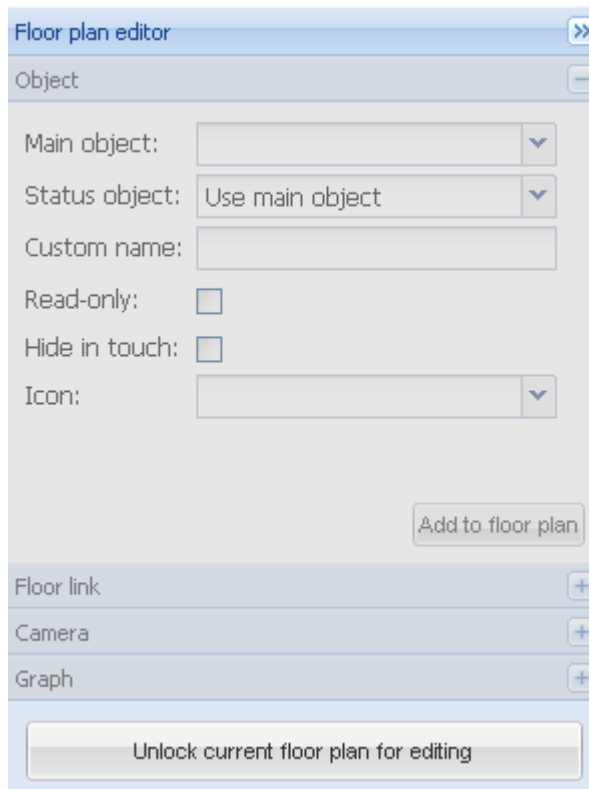
Для загрузки можно использовать изображения в следующих форматах: BMP, GIF, JPEG и PNG. Для удаления изображения, загруженного для соответствующего этажа, нужно не заполнять поле загрузки. Для удаления выбрать сохраненный этаж и нажать .

## Визуализация

После ввода параметров здания и этажей во окне *Buildings* во вкладке *Visualization* появляется графическое отображение здания. Здесь мы можем добавлять и расставлять иконки объектов по этажам зданий.



Нажав , вы можете убрать боковые вкладки; этим увеличив область отображения. Данная функция в особенности полезна при просмотре на небольших экранах.



## Редактор плана этажа здания

Справа располагается окно *Floor plan editor*. Нажав на *Unlock current floor plan for editing* вы можете видеть пять пунктов меню:

*Object* – добавление нового объекта или редактирование существующего.

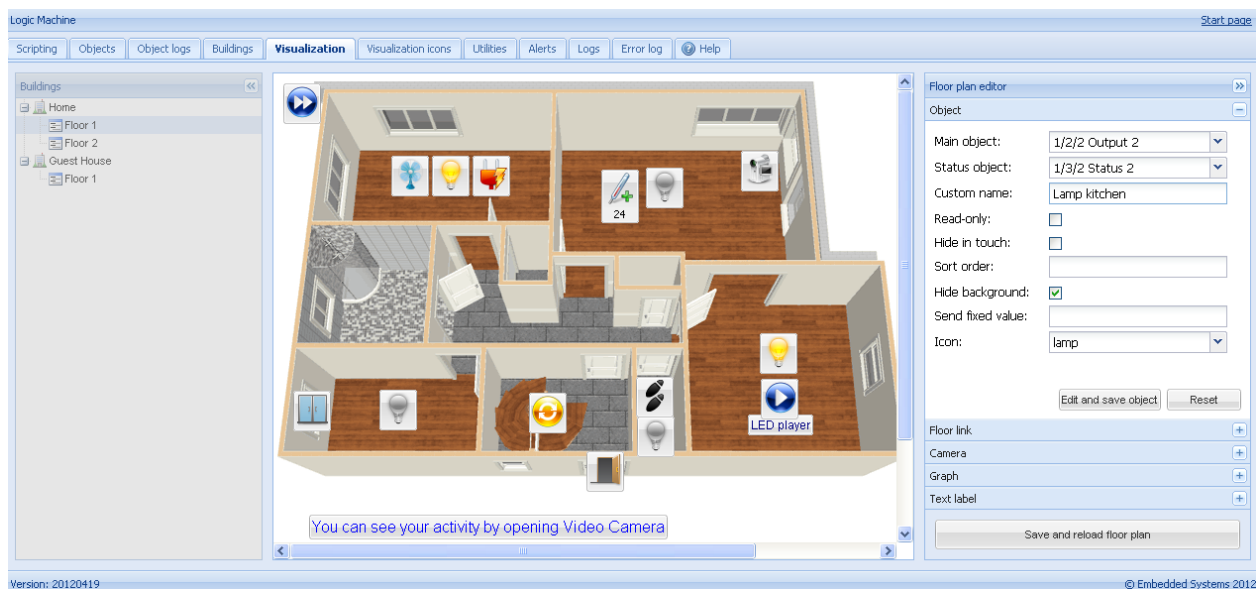
*Floor link* – связывание нескольких этажей (используется специальная пиктограмма).

*Camera* – настройка параметров цифровой IP камеры для отображение на плане этажа.

*Graph* – график (в реальном времени) контроля значений объектов.

*Text label* – надпись

## Объекты системы управления (Object)



*Main object* – список существующих групповых адресов шины KNX/EIB и других объектов; данные адреса доступны для редактирования во вкладке *Objects*.

*Status object* – список статусных объектов KNX/EIB.

*Custom name* – имя объекта.

*Read-only* – объект доступен только для чтения; функция установки значения недоступна.

*Hide in touch* – не отображать объект в *Touch Visualization* (не показывать на мобильных устройствах).

*Sort order* – порядок сортировки в *Touch Visualization*

*Hide background* – скрывать фон пиктограммы

*Send fixed value* – посылать выбранное значение объекту при нажатии на иконку объекта.

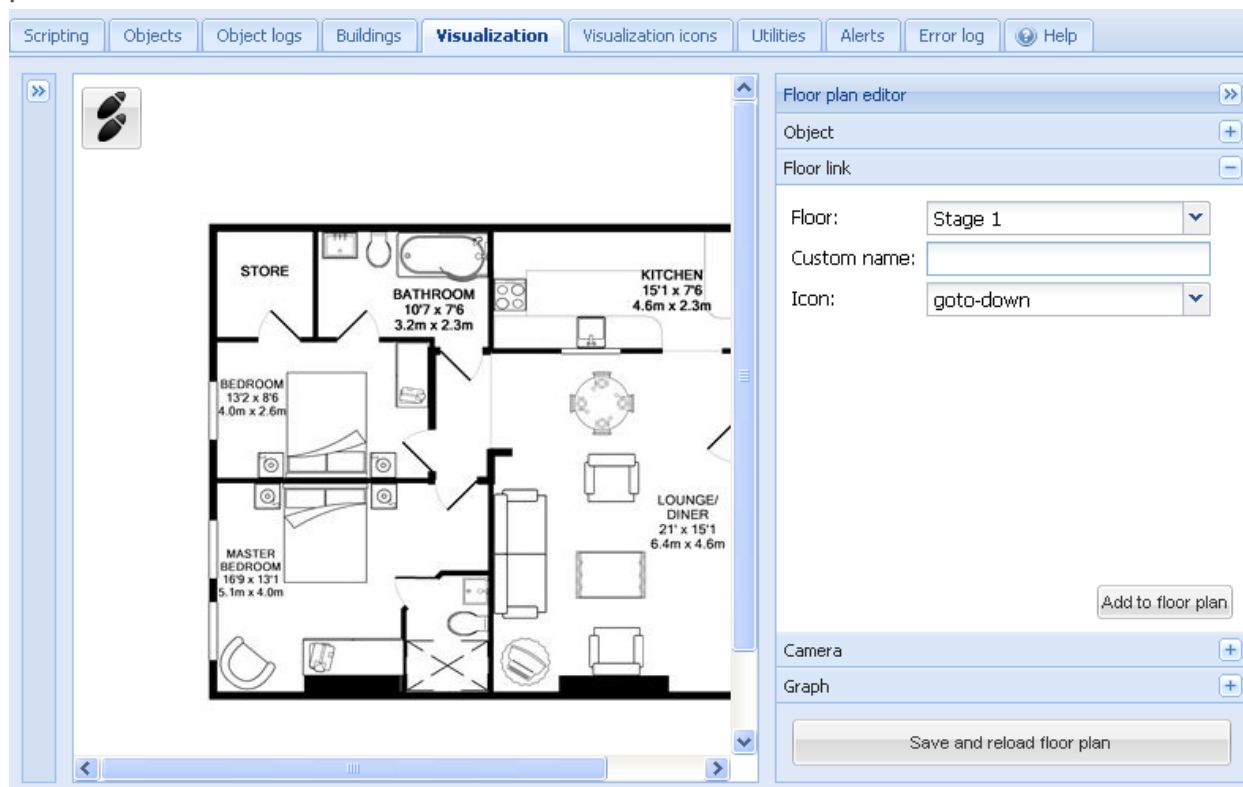
*Icon* – перечень доступных пиктограмм (вкладка *Visualization icons*).

После задания параметров объекта нужно нажать на кнопку *Add to floor plan*; на плане появится пиктограмма созданного объекта. Пользователь может переместить пиктограмму в любое место на плане. Следует помнить, что в режиме редактирования объект выключен. После добавления всех требуемых объектов нужно нажать *Save and reload floor plan*; объекты будут включены.

Для редактирования объектов нужно войти в режим редактирования (нажав на кнопку *Unlock current floor plan for editing*)

## Связывание этажей (*Floor link*)

Для более наглядного графического представления здания используется функция связывания этажей. Пользователь может добавить на план специальные пиктограммы, представляющие собой ссылки на планы других этажей



*Floor* – название этажа, к которому относится ссылка.

*Custom name* – имя ссылки.

*Hide background* – скрыть фон пиктограммы

*Icon* – отображаемая на карте пиктограмма.

После ввода параметров для связывания этажей, нужно нажать кнопку *Add to floor plan*; на плане появиться пиктограмма созданного объекта. Пользователь может переместить пиктограмму в любое место на плане. Следует помнить, что в режиме редактирования объект выключен. После добавления всех требуемых объектов нужно нажать *Save and reload floor plan*; объекты будут включены.

## Камера (Camera)

Logic Machine 2 поддерживает сетевые IP веб камеры сторонних производителей



*Source url* – URL видеокamеры.

*Width* – ширина окна для просмотра изображения.

*Height* – высота окна для просмотра изображения.

*Custom name* – имя объекта.

*Hide background* – скрыть фон пиктограммы

*Sort order* – порядок сортировки камер на мобильных устройствах

После ввода параметров работы видеокamеры нужно нажать на кнопку *Add to floor plan*; после чего появиться пиктограмма видеокamеры на плане. Пользователь может переместить пиктограмму в любое место на плане. Следует помнить, что в режиме редактирования объект выключен. После добавления всех требуемых объектов нужно нажать *Save and reload floor plan*; после этого объекты будут включены. Для вывода окна с картинкой, получаемой с IP веб-камеры, нужно нажать на пиктограмму видеокamеры. Окно можно двигать таким образом, чтобы не загромождать другие объекты на плане.

## График (Graph)

Для контроля текущих и предыдущих значений объектов в рамках визуализации используются графики реального времени.

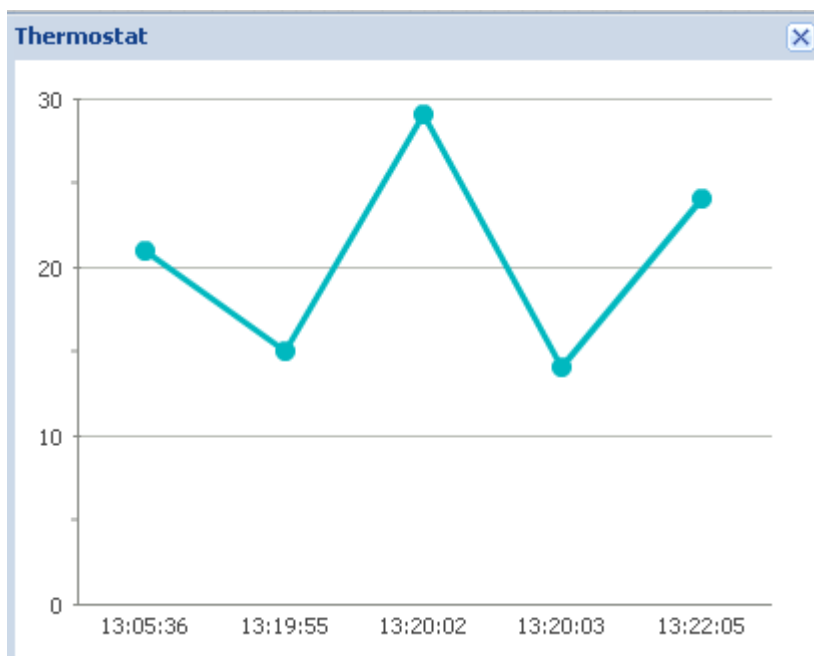
*Data object* – групповой адрес объекта.

*Custom name* – имя объекта.

*Icon* – пиктограмма для включения функции отображения графика.

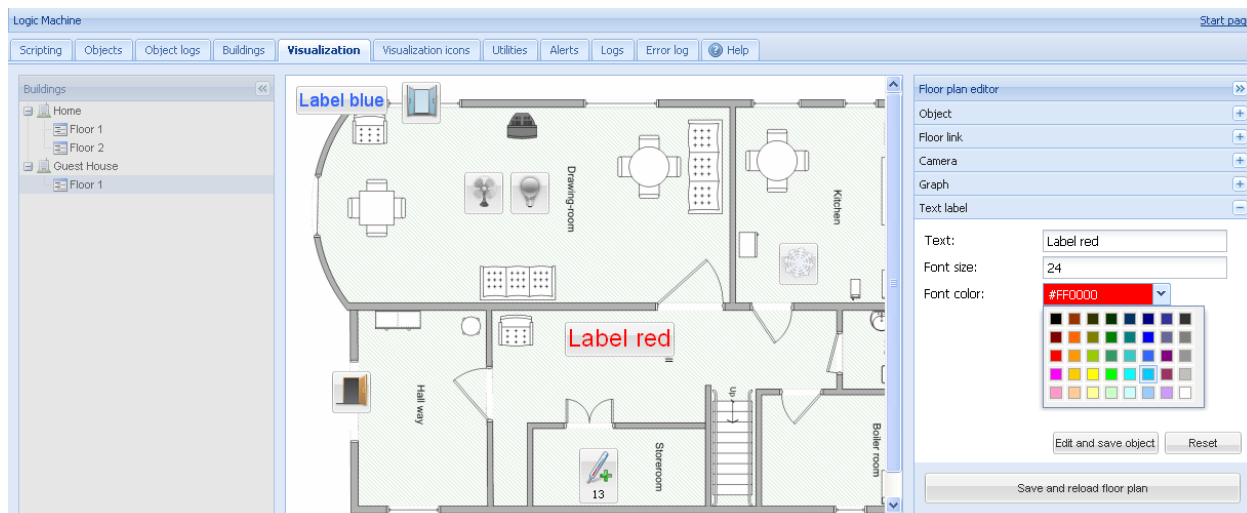
*Width* – ширина подокна для отображения графика.

*Height* – высота подокна для отображения графика.



После ввода параметров графика нужно нажать кнопку *Add to floor plan*; пиктограмма графика появится на плане. Пользователь может переместить пиктограмму в любое место плана. После добавления всех графиков, нужно нажать кнопку *Save and reload floor plan*;

## Надпись (Text Label)



*Text* – Текст надписи

*Font size* – размер шрифта

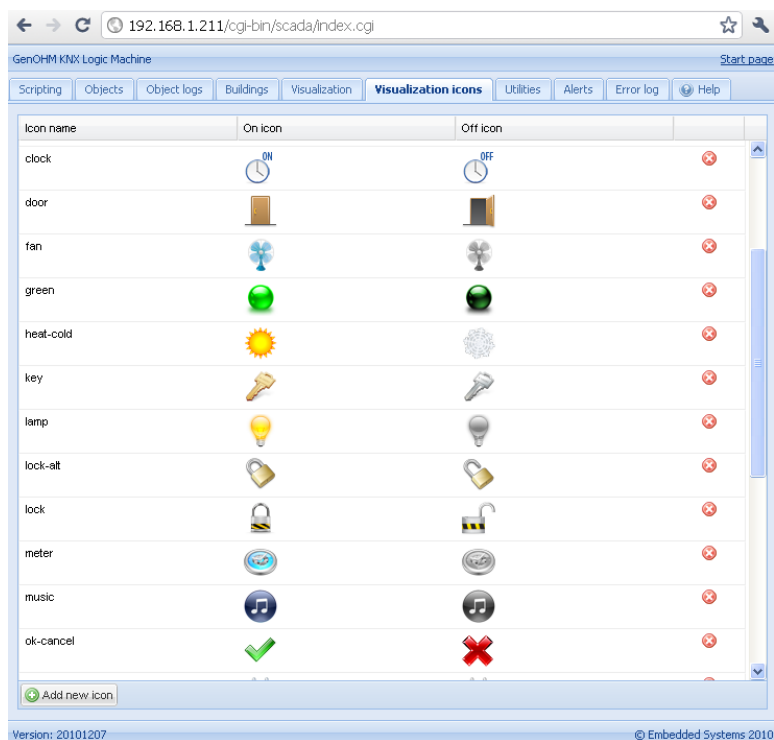
*Font color* – цвет шрифта (выбирается из палитры)

После ввода параметров надписи нужно нажать кнопку *Add to floor plan*; надпись появится на плане. Пользователь может переместить надпись в любое место плана. После добавления всех надписей, нужно нажать кнопку *Save and reload floor plan*;

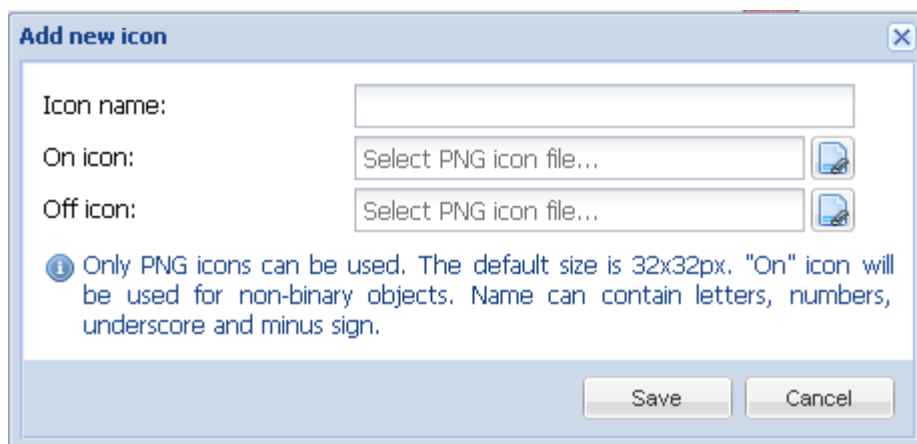


## Иконки, используемые при создании визуализации (Visualization icons)

Для просмотра перечня заранее установленных пиктограмм нужно открыть вкладку *Visualization icons*.



Для добавления новой пиктограммы, нужно нажать *Add new icon* и ввести требуемые параметры в появившемся окне.



По умолчанию размер иконки: 32 x 32 пикселей, размер можно изменить. Поддерживается анимация в файлах формата GIF.

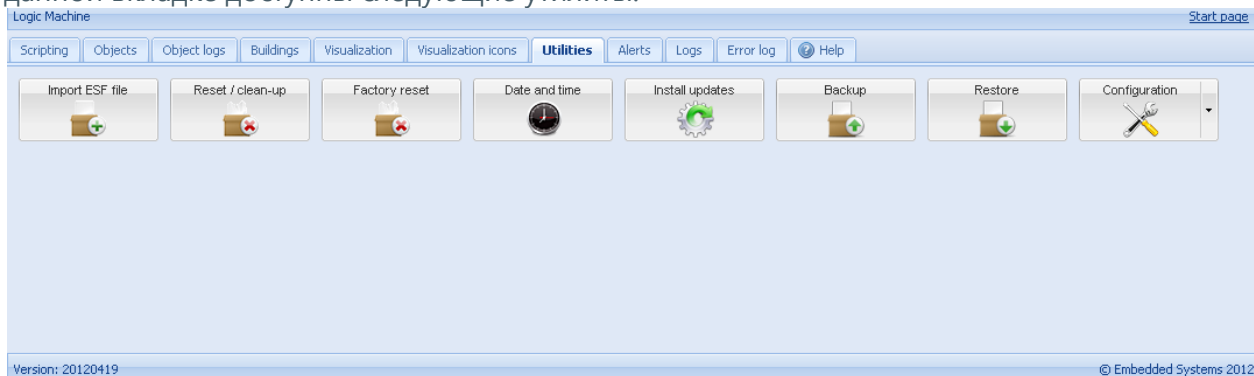
*Icon name* – название пиктограммы, отображаемое в перечне после добавления нового объекта. Допускается использовать буквенные символы, цифровые значения, нижнее подчеркивание и знак минус.

*On icon* – пиктограмма для отображения статуса «включен» для объектов, имеющих два состояния; эта пиктограмма будет использоваться для не бинарных объектов.

*Off icon* – пиктограмма для отображения статуса объекта «выключен». Только для бинарных объектов.

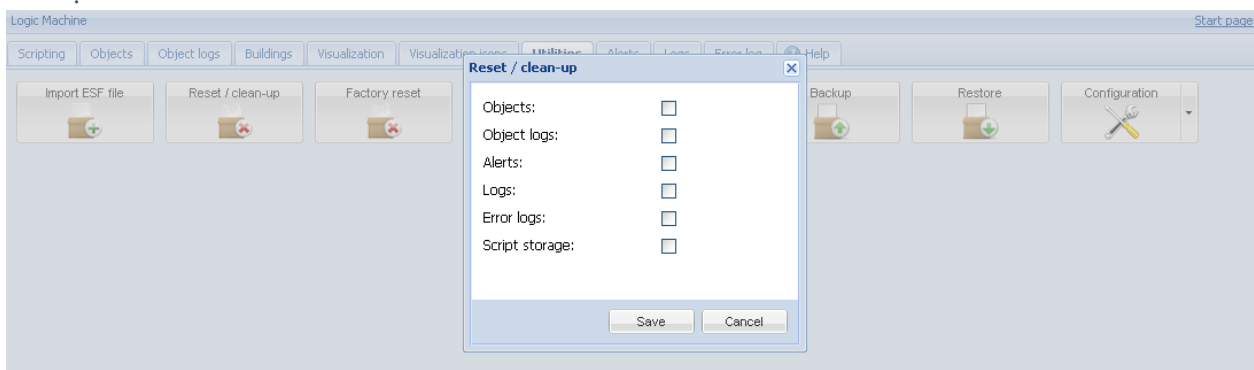
## Утилиты (Utilities)

В данной вкладке доступны следующие утилиты:



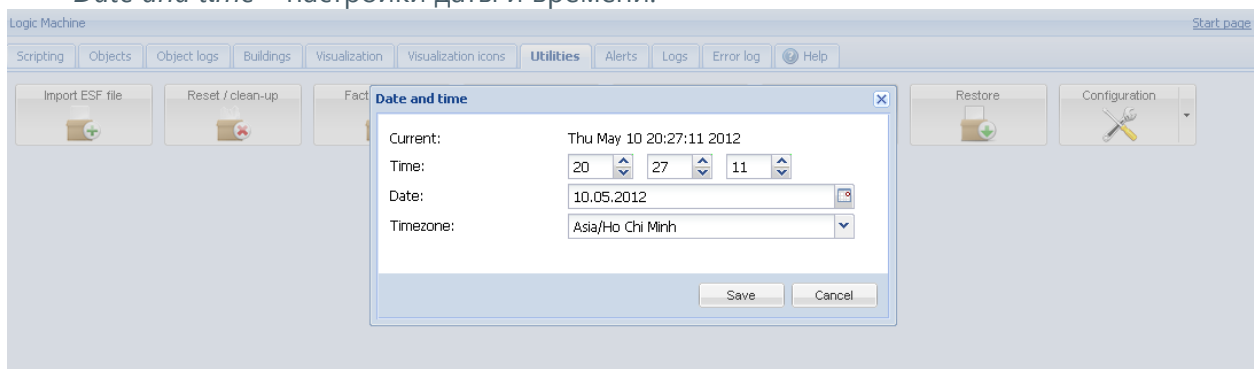
*Import ESF file* – импорт файла объектов ETS. Для корректной загрузки объектов следует задавать типы данных для импортируемых объектов. Существующие объекты не перезаписываются. Объекты, имеющие одно имя, считаются копиями и не импортируются.

*Reset object DB* – удалить все объекты из памяти Logic Machine 2; объекты также удаляются и с визуализации



*Factory reset* – удаление всех пользовательских настроек и возврат к заводским настройкам

*Date and time* – настройки даты и времени.



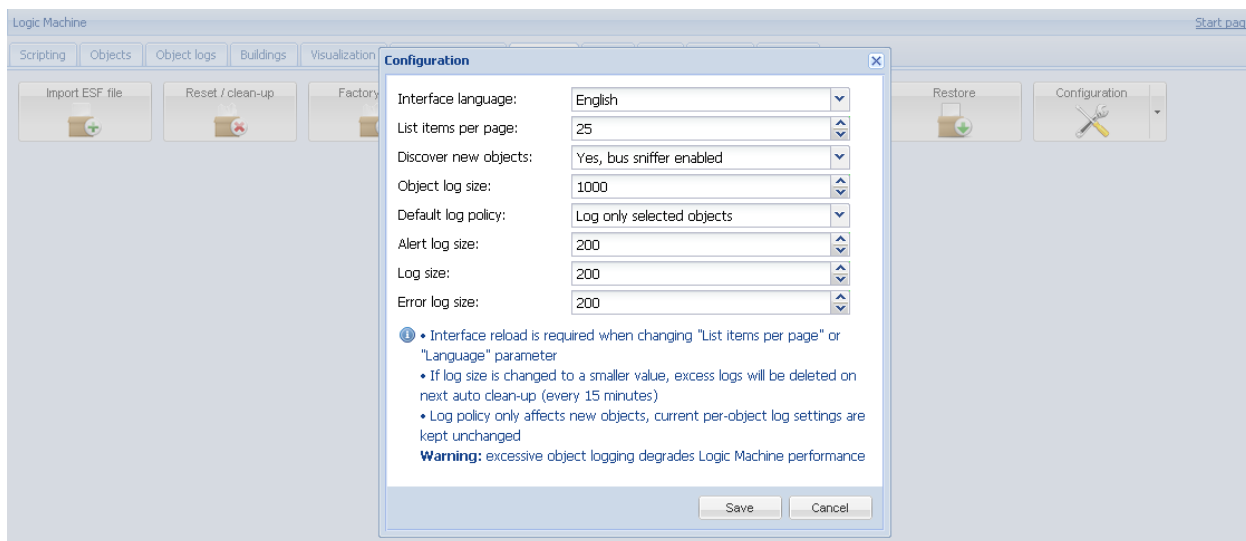
*Install updates* – загрузить файл обновления Logic Machine 2 \*.Imu. Logic Machine 2 выполняет перезагрузку после успешного обновления.



*Backup* – резервное копирование всех объектов, журналов и скриптов. Планы этажей резервному копированию не подлежат.

*Restore* – восстановление данных из backup

*Configuration* – при нажатии на стрелку, вы можете войти в настройки KNX Connection и User Access. Если нажать на саму кнопку Configuration, появятся окно с системными настройками.



**Interface language** – язык интерфейса.

**List items per page** – кол-во строк на странице для модулей Objects, Object logs, Alerts и так далее.

**Discover new objects** – в любом случае снифер KNX объектов будет работать. Если выбрано yes, то все обнаруженные объекты будут добавляться в модуль Objects

**Object log size** – максимальное кол-во записей в журнале объектов (Object logs)

**Default log policy** – в журнал сохраняются события либо от всех объектов либо только от тех объектов где установлен такой флаг

**Alert log size** – максимальное кол-во предупреждений которое будет храниться в журнале (Alert)

**Log size** – максимальное кол-во записей функции log которое будет храниться в журнале (Log)

**Error log size** – максимальное кол-во ошибок которое будет храниться в журнале (Error)

**Внимание!** Если вы изменили параметры "List items per page" или "Language" то требуется перезагрузить интерфейс LM в браузере (кнопка Refresh)

**Внимание!** Если новое значение размера журнала меньше чем кол-во записей журнала, то лишние будут удалены процедурой auto clean-up (которая выполняется каждые 15 минут)

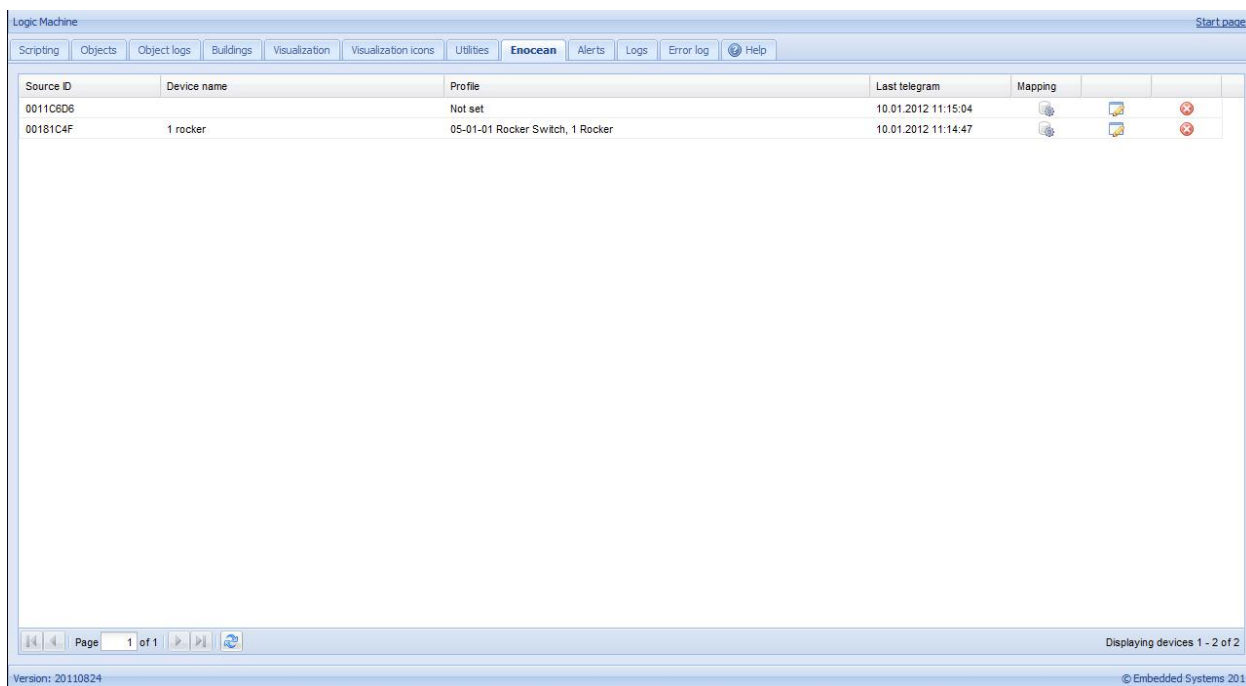
**Внимание!** Изменение параметра **Default log policy** будет влиять на запись только новых событий, предыдущие останутся без изменений

**Предупреждение!** Включение параметра *Default log policy* на автоматическое сохранение всех объектов сильно уменьшит производительность Logic Machine

## EnOcean

Существует возможность организации беспроводного управления KNX сетью.

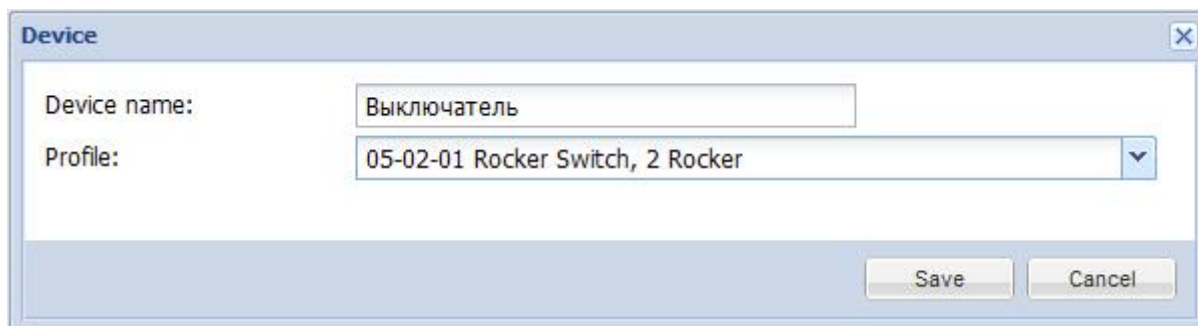
Во вкладке EnOcean (зайти в Logic Machine > EnOcean) отображаются элементы управления, работающие по радио-каналу.




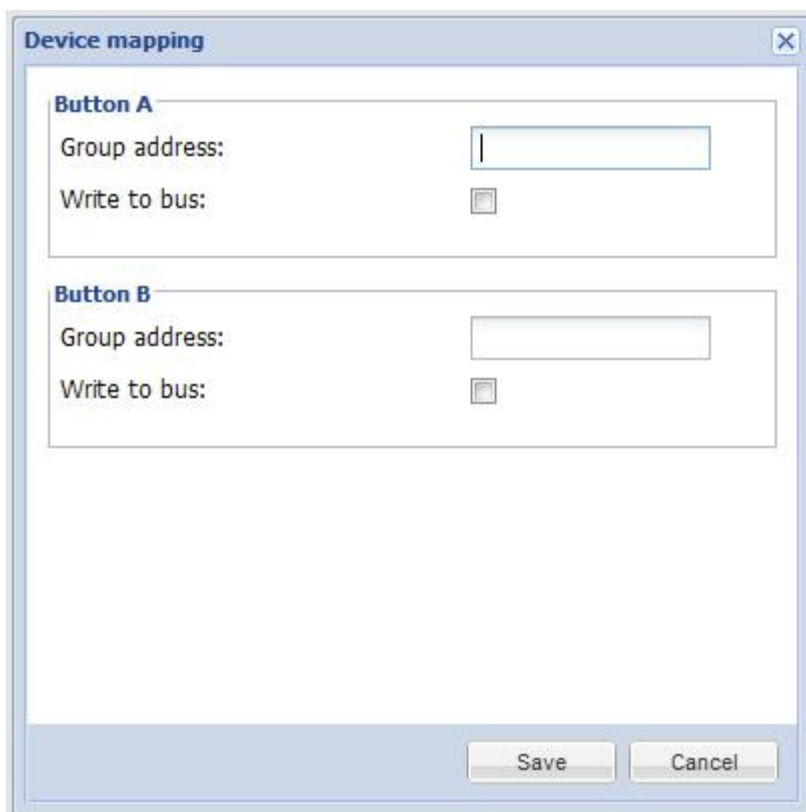
Здесь отображаются устройства EnOcean с которых были отправлены телеграммы.

Зеленым цветом выделены устройства, передавшие телеграмму.

При нажатии на строку открывается окно настройки устройства.



При нажатии кнопки  происходит привязка канала EnOcean устройства к групповому адресу.



The 'Device mapping' dialog box contains two sections, 'Button A' and 'Button B'. Each section has a 'Group address' text box and a 'Write to bus' checkbox. At the bottom are 'Save' and 'Cancel' buttons.

Section	Group address	Write to bus
Button A	<input type="text"/>	<input type="checkbox"/>
Button B	<input type="text"/>	<input type="checkbox"/>

## Предупредительные сообщения (Alerts)

Во вкладке *Alert* содержится перечень заданных с использованием функции **alert** предупредительных сообщений для соответствующих скриптов. Сообщения хранятся на карте памяти.

Alert time	Message
01.01.1970 10:20:42	read error
01.01.1970 10:20:22	read error
01.01.1970 10:20:02	read error
01.01.1970 10:12:58	read error

Page 1 of 93

Displaying alerts 1 - 25 of 2317

Используя панель управления, пользователь имеет возможность перемещаться и перезагружать страницы.



имеет

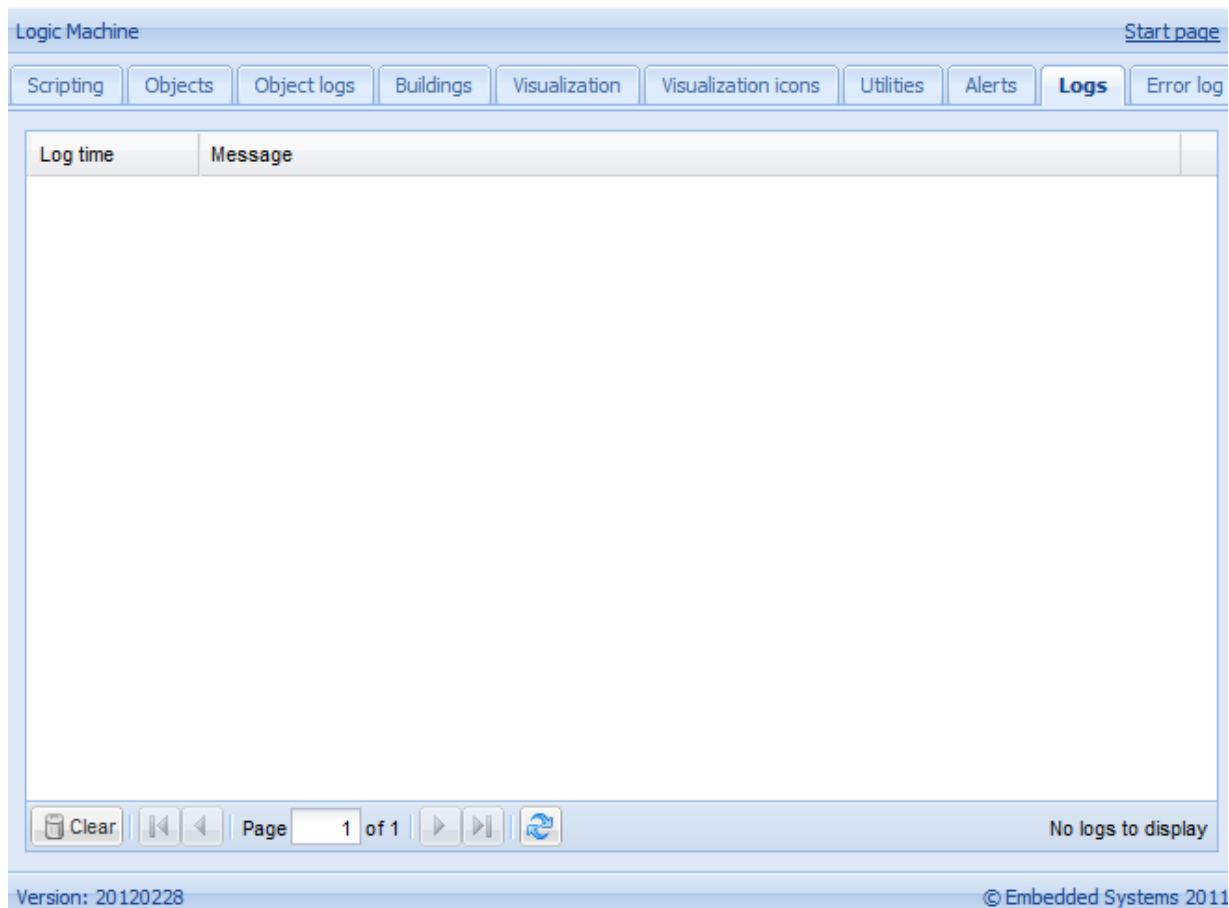
Предупредительное сообщение (пример):

```
1.temperature = 25.3
2.
```

```
3. if temperature > 24 then
4. -- результирующее сообщение: 'Temperature levels are too high: 25.3'
5. alert('Temperature level is too high: %.1f', temperature)
6. end
```

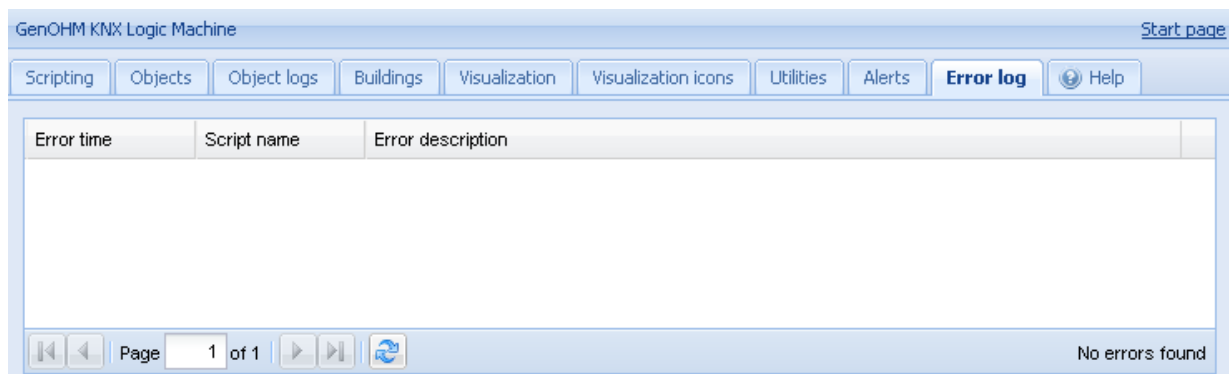
## Журнал (Logs)

Во вкладке *Logs* содержится перечень заданных с использованием функции `log` сообщений для соответствующих скриптов. Обычно используется при отладке скриптов



## Журнал ошибок (Error logs)

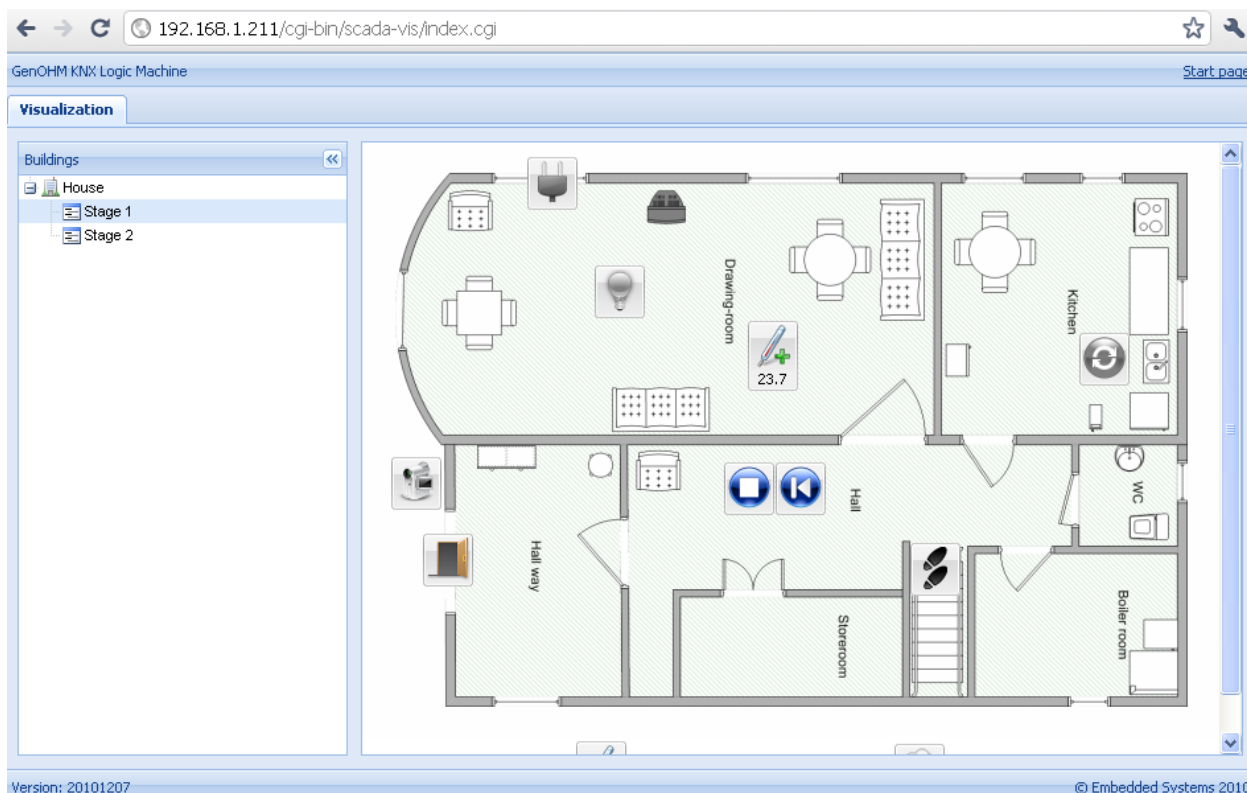
Сообщения об ошибках, обнаруженных в скриптах, отображаются во вкладке *Error log*.



## Справка

Во вкладке *Help* содержится справочная информация по синтаксису скриптов.

# WEBI User mode visualization (Управление объектами визуализации, нормальный)



## Touch visualization – режим графического представления данных с использованием сенсорных устройств

Данный режим предназначен для сенсорных устройств (iPhone/iPod/iPad/Android). Все объекты конфигурации *Logic Machine 2* по умолчанию доступны в данном режиме (если не включена функция *Hide in touch*).

*Logic Machine 2* поддерживает три уровня доступа: чтение, редактирование и администрирование.

Уровень доступа	Имя пользователя	Пароль
Только чтение	visview	visview
Редактирование	viscontrol	viscontrol
Редактирование с правами администратора	visadmin	visadmin

Главное окно – Building; пользователь выбирает этаж и здание для внесения изменений (просмотра). После этого на экран выводится перечень всех объектов, связанных с выбранным этажом; управление осуществляется при помощи круглой пиктограммы справа или слева (для двоичных объектов) или направляющей (для масштабированных объектов).

## ***WEBI Touch visualization (Управление объектами визуализации мобильный)***



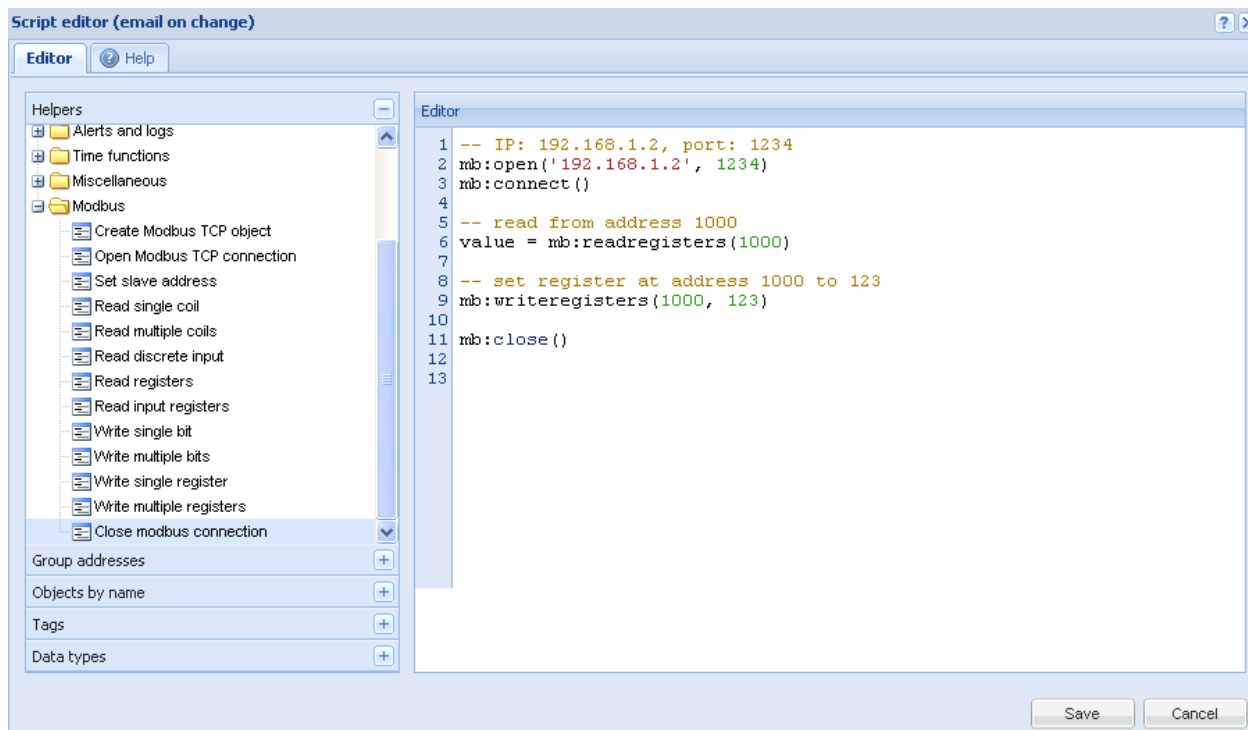
### Настройка визуализации на сенсорном устройстве (iPad в этом случае)

- Проверьте есть ли беспроводное соединение между вашим iPad и Logic Machine (через точку доступа).
- Введите в браузере IP адрес Logic Machine (по умолчанию 192.168.0.10).
- Нажмите на пиктограмму Touch Visualization.
- Сохраните ярлык на домашнем экране Вашего iPad

### **Взаимодействие Modbus RTU/TCP с LM2**

Modbus RTU поддерживается через интерфейс RS485. Modbus TCP поддерживается через интерфейс Ethernet. Взаимодействие с Modbus происходит с помощью скриптов (обычно используется резидентный скрипт который читает значение переменных Modbus через определенный промежуток времени и сохраняет в KNX объект).

После того как создан скрипт вы можете добавить код в Script Editor. Для вашего удобства вы можете воспользоваться множеством предопределенных фрагментов кода скриптов в окне Helpers, которые вы можете вставить в свой скрипт.



## Master functions

### *mb:setslave(slaveid)*

установка slaveid для чтения/ записи данных

### *mb:readcoils(start, count)*

### *mb:readdiscreteinputs(start, count)*

### *mb:readregisters(start, count)*

### *mb:readinputregisters(start, count)*

прочитать кол-во(count) регистров (registers/coils) начиная со стартового адреса (start). При успешном выполнении возвращаются все значения иначе nil и описание ошибки

### *mb:writebits(start, v1, [v2, [v3, ...]])*

### *mb:writeregisters(start, v1, [v2, [v3, ...]])*

записать значения в регистры(registers/coils)

начиная со стартового адреса (start) в случае ошибки возвращает nil и описание ошибки

### *mb:reportslaveid()*

читает внутренние данные подчиненного (slave) устройства. В случае ошибки возвращает nil и описание ошибки

```
-- init modbus on first script execution
```

```
if not mb then
```

```
    require('luamodbus')
```

```
    mb = luamodbus.tcp()
```

```
end
```



```
-- prepare connection to given ip and port
mb:open('192.168.1.100', 1502)

-- open connection and check the result
if mb:connect() then
    -- read 3 input registers, function returns 3 variables
    local x, y, z = mb:readinputregisters(1, 3)
    -- first variable will be nil if read failed
    if x then
        local message = string.format('1: %d; 2: %d; 3: %d', x, y, z)
        alert(message)
    end
    -- end session
    mb:close()
else
    alert('connection failed')
end
```

## Визуализация Modbus объектов

Используйте `grp.write()` для связывания объектов Modbus с KNX объектами и используйте новый KNX объект для визуализации

## Пример использования (residentscript)

Задача: прочитать три регистра из Modbus TCP и записать результат в Alerts.

```
1. -- инициализация modbus при первом запуске скрипта
2.  if not mb then
3.      require('luamodbus')
4.      mb = luamodbus.tcp()
5.  end
6.
7.  -- подготавливаем соединение по заданному адресу и порту
8.  mb:open('192.168.1.100', 1502)
9.
10. -- устанавливаем соединение и проверяем результат
11. if mb:connect() then
12.     -- читаем 3 входных регистра, функция возвращает 3 переменные
13.     local x, y, z = mb:readinputregisters(1, 3)
14.
15.     -- первая переменная будет равно nil если операция чтения не отработала
16.     if x then
17.         local message = string.format('1: %d; 2: %d; 3: %d', x, y, z)
18.         alert(message)
19.     end
20.
21.     -- закрываем соединение
```

```
22.     mb:close()  
23. else  
24.     alert('connection failed')  
25. end
```

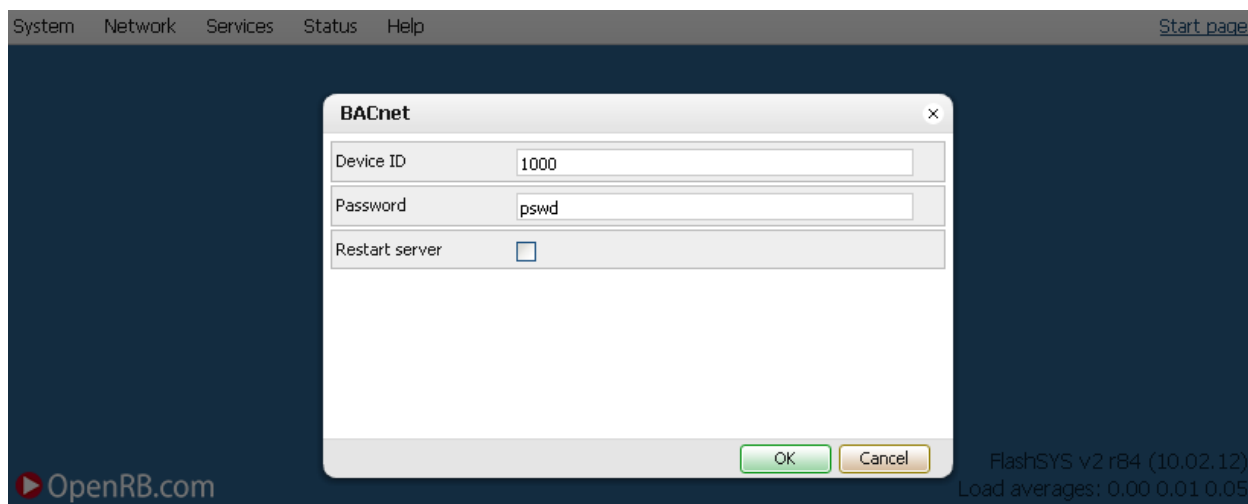
**Задача:** Прочитать два параметра (напряжение и силу тока) из Modbus мультиметра и сохранить данные в объекте KNX. Будьте внимательны соединяя LM2 с Modbus устройством, RS485 А нужно соединить с минусом, RS485 В соединить с плюсом.

```
1.-- инициализируем Modbus при первом запуске скрипта  
2.if not mb then  
3.    require('luamodbus')  
4.    mb = luamodbus.rtu()  
5.    mb:open('/dev/ttyS2', 9600, 'E', 8, 1, 'H')  
6.    mb:connect()  
7.end  
8.  
9.-- устанавливаем slave ID для чтения и записи данных  
10.mb:setslave(20)  
11.  
12.-- читаем значение напряжения из 32-bit регистра  
13.r1, r2 = mb:readregisters(0x1000, 2)  
14.result = bit.lshift(r1, 16) + r2  
15.grp.write('5/5/1',result)  
16.  
17.-- читаем значение силы тока из 32-bit регистра  
18.r1, r2 = mb:readregisters(0x100E, 2)  
19.result = bit.lshift(r1, 16) + r2  
20.grp.write('5/5/2',result)
```

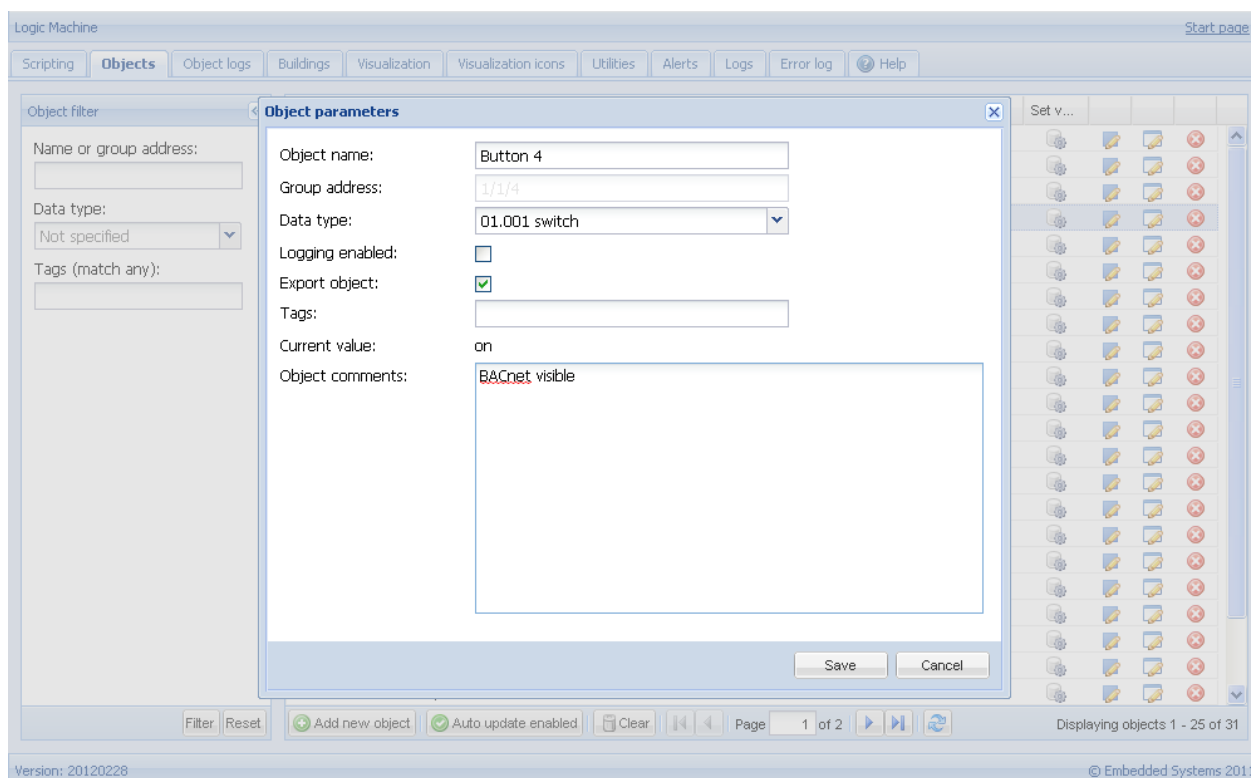
## Взаимодействие BACnet IP с LM2

Определите BACnet *Device ID* и *Password* см. FlashSys\Network\BACnet (подключение к сети BACnet) (стр. 31).

(обычно используется для удаленной перезагрузки) в **Network Configuration** → **Network** → **BACnet**.



Установите флаг Export object для тех объектов KNX/EIB в Logic Machine которые должны быть видны в сети BACnet для операций чтения и записи. Двоичные объекты будут видны как Binary Values, другие числовые типы как Analog Values. Другие типы в данный момент не поддерживаются. Запись в шину KNX управляется через свойство Relinquish Default property



**Внимание!** При перезагрузке сервиса BACnet будет посланы запросы Reinitialize Device которые перезагрузят все объекты, список приоритетов будет установлен в NULL.

## Взаимодействие со встроенными портами ввода- вывода

	Бинарные/Аналоговые входы	Бинарные выходы
Logic Machine 2 Control	4	4
Logic Machine 2 Interface	1	1

Прочитать статус бинарного выхода номер 1

```
status = digital.read(1)
```

Записать в выход номер 1 значение “1”

```
digital.write(1, true)
```

Прочитать значение из аналогового входа номер 2

```
value = analog.read(2)
```

Прочитать значение напряжения из аналогового входа номер 2

```
value = analog.voltread(2)
```

Прочитать значение из аналогового входа номер 2, и если напряжение меньше 5V вернуть false иначе true

```
value = analog.binread(2)
```

## Взаимодействие с последовательным портом RS232

### Открыть соединение RS232

```
1.require('serial')
2.port = serial.open('/dev/ttyS2', {
3.  baudrate = 38400,
4.  databits = 8,
5.  stopbits = 1,
6.  parity = 'even',
7.  duplex = 'half'
8.})
```

### Запись в порт

```
port:write('test data')
```

### Чтение с блокировкой (скрипт будет ждать пока не прочитает 10 символов)

```
data = port:read(10)
```

### Чтение с прерыванием по времени (Timeout) (скрипт будет ждать 10 символов 20 секунд)

```
data = port:read(10, 20)
```

### Закреть соединение

```
port:close()
```

# ***WEBI User mode Shedulers\_ (Управление запланированными событиями системы)***

---

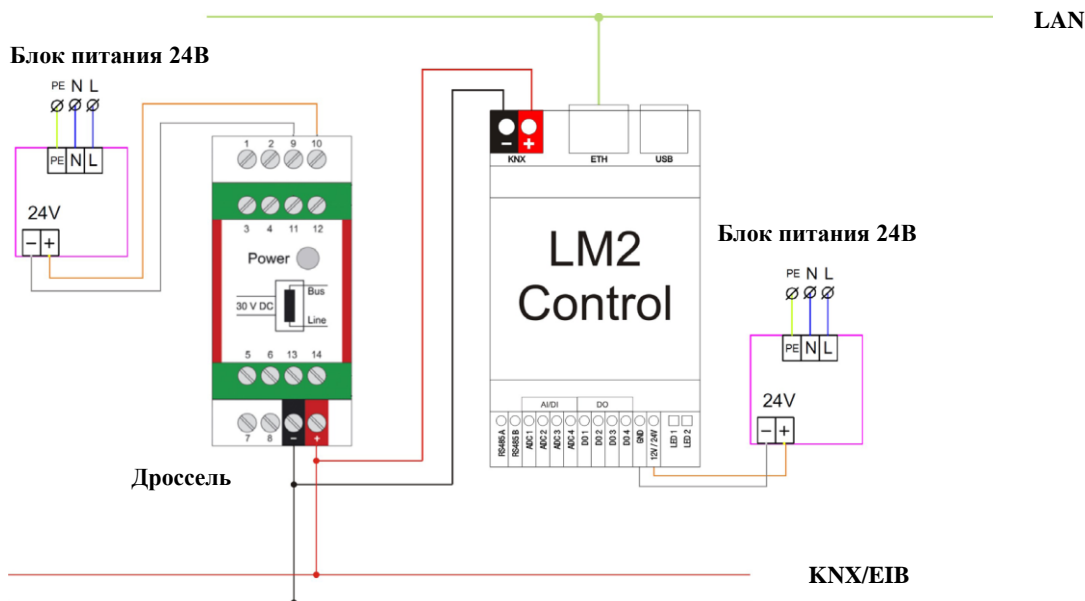
Раздел в разработке

# ***WEBI Trend log (Просмотр стандартных отчётов системы)***

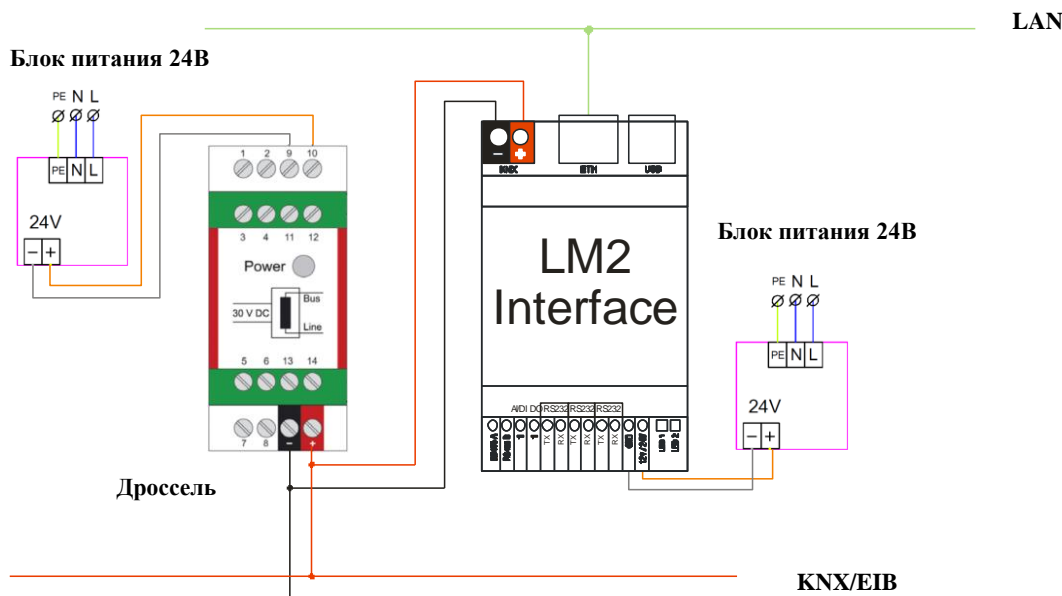
---

Раздел в разработке

## Подключение Logic Machine 2 Control к LAN и шине KNX



## Подключение Logic Machine 2 Interface к LAN и шине KNX



- Управление устройствами, подключенными к шину KNX
- Визуализация, SCADA
- Управление с помощью скриптов

L	L
N	N
PE	PE

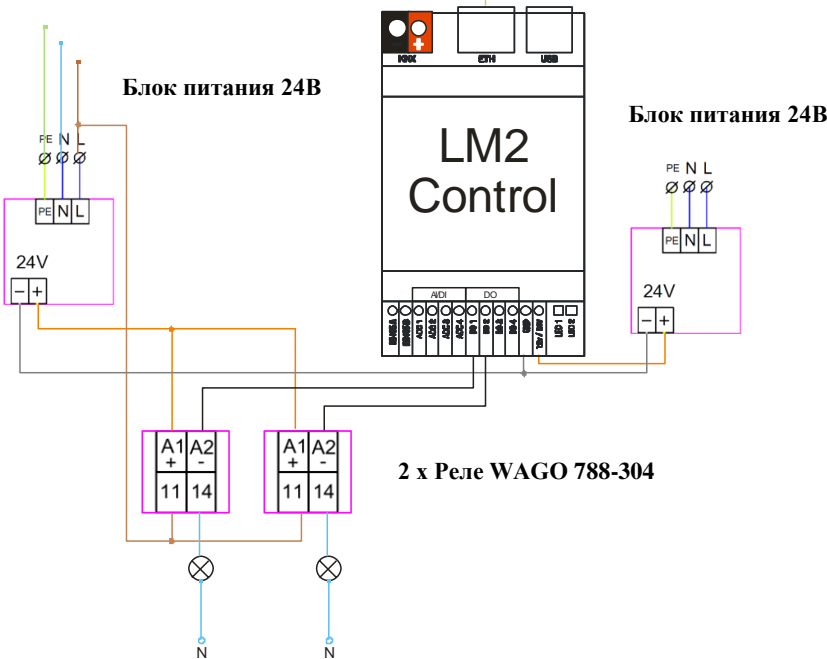
Для работы EVIKA Logic Machine 2 необходимо подать напряжение от 5 до 36 вольт (в комплекте идёт блок питания на 24В) на клеммы 12/24В и GND. Для работы шины KNX подключаем блок питания через EVIKA KNXCHOKE. Шина KNX подключается с помощью клемм WAGO (имеются в комплекте) на соответствующее место. Подключение к локальной сети производится с помощью стандартного разъёма RJ-45.





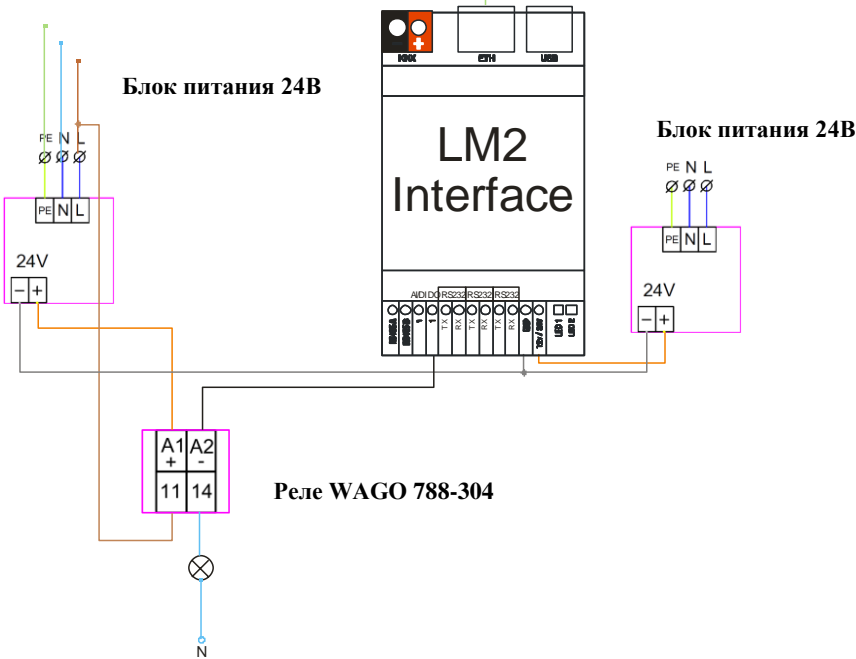
Подключение реле WAGO 788-304 к Logic Machine 2 Control

LAN




Подключение реле WAGO 788-304 к Logic Machine 2 Control

LAN



Функции:  
Включить/выключить нагрузку

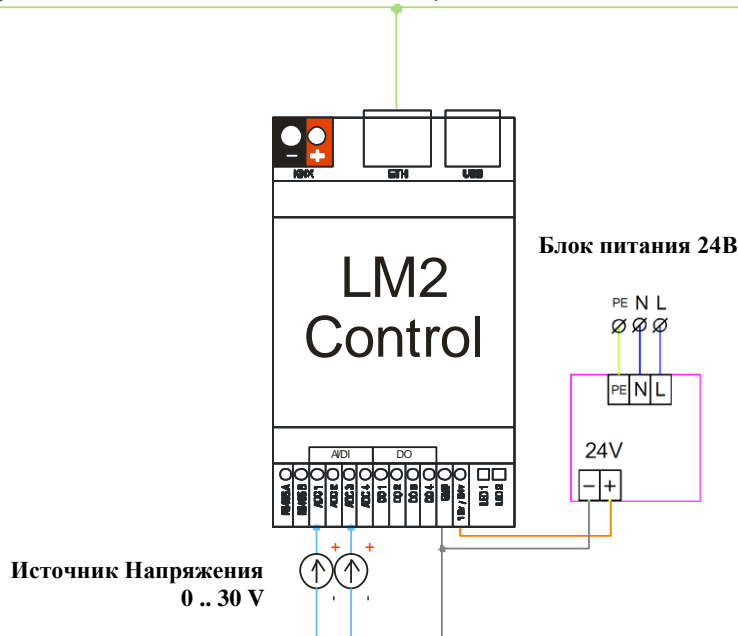
Для подключения релейного модуля необходим дополнительный блок питания на 24В. Минусовая клемма 24В дополнительного блока питания соединяется с минусовой клеммой блока питания Logic Machine2. Плюсовая клемма 24В блока питания заводится на клемму A1 релейных модулей, клемма A2 релейного модуля заводится на соответствующий канал EVIKA Logic

L-	фаза
N –	нейтраль
PE –	земля
 -	нагрузка (светильник)
A1,A2	управляющие

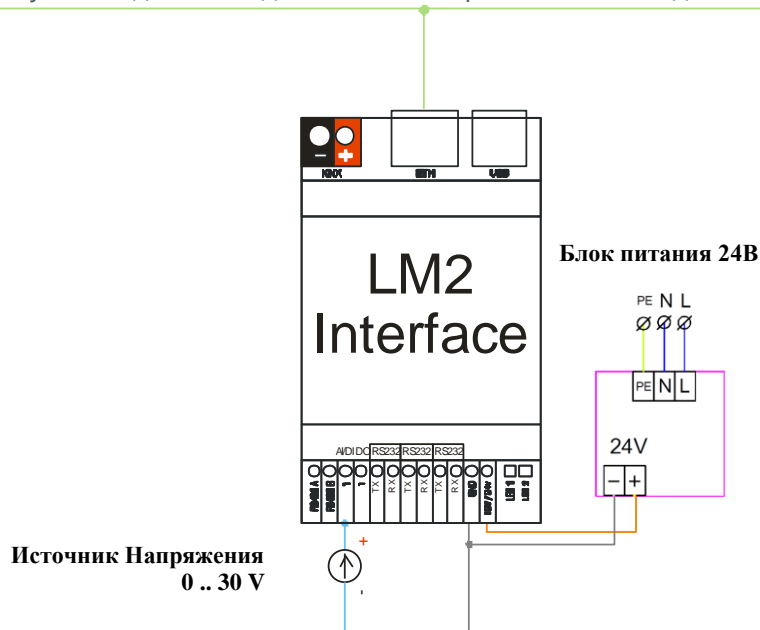
Machine2 (в нашем примере выходы DO1 и DO2).

	клеммы реле WAGO 788-304
11, 14	силовые клеммы реле WAGO 788-304

LAN



LAN



Для подключения датчика 0-10В (возможно до 30В) необходимо одну клемму выключателя соединить минусовой клеммой блока питания Logic Machine2, а другую с входом EVIKA Logic Machine2 (в нашем примере вход ADC1, ADC3).