

IoT-платформа Home Assistant

Программирование шлюзов Reallab в IoT-платформе Home Assistant

Программирование шлюзов Reallab в IoT-платформе Home Assistant

Руководство пользователя

© НИЛ АП, 2026

Версия от 31 августа 2026 г.

Одной проблемой стало меньше!

Уважаемый покупатель!

Научно-исследовательская лаборатория автоматизации проектирования (НИЛ АП) благодарит Вас за покупку и просит сообщать нам свои пожелания по улучшению этого руководства или описанной в нем продукции. Ваши пожелания можно направлять по почтовому или электронному адресу, а также сообщать по телефону:

НИЛ АП, пер. Биржевой спуск, 8, Таганрог, 347900,

Тел.: (495) 26-66-700,

e-mail: info@reallab.ru, www.reallab.ru

Вы можете также получить консультации по применению нашей продукции, воспользовавшись указанными выше координатами.

Пожалуйста, внимательно изучите настоящее руководство. Это позволит вам в кратчайший срок и наилучшим образом использовать приобретенное изделие.

Авторские права на программное обеспечение, устройства и настоящее руководство принадлежат НИЛ АП.
--

Оглавление

1. Вводная часть	6
1.1. Введение	6
1.2. Общие сведения о Home Assistant	6
1.3. Преимущества Home Assistant перед другими IoT-платформами.....	8
1.4. Шлюзы RealLab поддерживающие Home Assistant	9
1.4.1. Модельный ряд промышленных IoT шлюзов Reallab	9
1.4.2. Основные характеристики шлюза	10
1.4.3. Программное обеспечение и поддерживаемые протоколы	10
1.4.4. Интерфейс шлюзов	12
1.4.5. Сетевые интерфейсы и настройка связи: Настройка Ethernet и Wi-Fi (Клиент / Точка доступа)	13
2. Начало работы	15
2.1. Подключение к шлюзу и запуск интерфейса Home Assistant	15
2.2. Доступ к веб-интерфейсу Home Assistant.....	17
2.2.1. Авторизация в Home Assistant	17
2.3. Работа с интерфейсом Home Assistant	21
2.3.1. Элементы боковой панели навигации.....	21
2.3.2. Элементы боковой панели навигации.....	22
2.3.3. Подробный разбор раздела «Настройки» пользователя.....	23
2.3.4. Инструменты разработчика	25
2.3.5. Структура сущностей (Entities) в Home Assistant	26
3. Подключение устройств. Работа с протоколами	27
3.1. MQTT	27
3.1.1. Подключение к MQTT брокеру	27
3.1.2. Публикация команд	30
3.2. Zigbee	33
3.2.1. Zigbee Home Automation настройка	34

3.2.2. Zigbee Home Automation подключение устройств	39
3.2.3. Zigbee2MQTT настройка	42
3.2.4. Zigbee2MQTT подключение устройств	44
3.3. LoRaWAN	50
3.3.1. Подключения LoRa датчика к ChirpStack	52
3.3.2. Получение и обработка данных LoRaWAN в Home Assistant через MQTT	53
3.4. Modbus.....	58
3.4.1. Подключение к шлюзу по SSH	59
3.4.2. Настройка Modbus RTU.....	62
3.4.3. Настройка Modbus TCP.....	64
3.4.4. Применение конфигурации	65
4. Программирование алгоритмов и автоматизация	65
4.1. Инструменты автоматизации в Home Assistant	65
4.1.1. Автоматизации (Automations)	65
4.1.2. Скрипты (Scripts).....	66
4.1.3. Сцены (Scenes).....	66
4.1.4. Чертежи (Blueprints).....	67
4.2. Создание автоматизаций: от простого к сложному.....	67
4.2.1. Визуальный редактор: триггеры, условия, действия.....	67
4.2.2. Основные типы триггеров	69
4.2.3. Режимы выполнения автоматизаций.....	70
4.2.4. Редактирование автоматизаций в YAML	71
4.3. Расширенная логика с Node-RED	72
4.3.1. Установка и настройка Node-RED и интеграции с Home Assistant	73
5. Визуализация в home assistant	74
5.1. Общие сведения о системе визуализации	74

5.2. Создание новой панели управления.....	74
5.3. Навигация и режимы работы с панелью	78
5.4. Добавление и настройка карточек.....	79
5.4.1. Добавление новой карточки.....	79
5.4.2. Пример добавления карточек	79
6. Работа с модулями ввода и вывода reallab	80
6.1. Модули серии NLS xx Ethernet с поддержкой MQTT	80
6.1.1. Предварительное конфигурирование модуля	80
6.1.2. Приём данных в Home Assistant	81
6.2. Модули с поддержкой Modbus RTU/TCP	81
6.2.1. Предварительное конфигурирование модулей	81
6.3. Модули с поддержкой LoRaWAN	83
7. Диагностика и устранение неисправностей.....	83
7.1. Системный мониторинг ресурсов шлюза	83
7.1.1. Просмотр информации о системе	84
7.1.2. Управление процессами	84
7.2. Просмотр системных журналов и диагностика ошибок	85
7.2.1. Общий подход к логированию	85
7.2.2. Рекомендации по анализу логов	85
7.3. Типовые неисправности и методы их устранения	86
Лист регистрации изменений	88

1. Вводная часть

1.1. Введение

Настоящее руководство по настройке и программированию содержит необходимую информацию для начала работы и программирования в IoT-платформе Home Assistant. Процесс ознакомления с IoT-платформой Home Assistant описан применительно к шлюзам RealLab серии NLS-Gateway и нескольким сенсорам, и модулям ввода и вывода, но часть информации в данном документе является универсальной и может быть использована для обучения работе с любыми контроллерами, программируемыми в этой среде.

Документ соответствует версии IoT-платформы Home Assistant:

- Core - 2026.5.4
- Frontend - 20260429.4

1.2. Общие сведения о Home Assistant

Home Assistant (HA) представляет собой открытую платформу для автоматизации и управления IoT-устройствами, обеспечивающую единую среду для настройки логики, мониторинга и интеграции различного оборудования. Основными частями системы являются:

- **Ядро (Home Assistant Core)**, отвечающее за логику и обработку событий
- **Веб-интерфейс (Frontend/Dashboard)** для управления и визуализации.

В Home Assistant предусмотрено несколько способов создания прикладного ПО: через визуальный редактор автоматизаций, с помощью скриптов или посредством прямой правки конфигурационных файлов на языке YAML. Платформа поддерживает модульную структуру, что позволяет реализовывать как простые алгоритмы «условие-действие», так и сложные сценарии с использованием шаблонов (Templates) и логических операторов.

В рамках одного сервера Home Assistant могут быть объединены устройства сотен различных производителей. Платформа выступает в роли центрального узла (хаба), обеспечивая взаимодействие между устройствами, работающими по разным протоколам (Modbus, MQTT, Zigbee, Wi-Fi, Bluetooth и др.). Каждое устройство представляется в системе в виде набора сущностей (Entities), которые могут взаимодействовать друг с другом независимо от их физического интерфейса.

Логика автоматизаций выполняется ядром системы в реальном времени. Встроенные средства проверки конфигурации и диагностические журналы позволяют выявлять ошибки как на этапе написания кода, так и в процессе работы системы, выдавая информативные сообщения о некорректных параметрах.

Для отладки программы без фактического изменения состояния оборудования предусмотрены «Инструменты разработчика». Они позволяют имитировать изменения состояний сущностей, тестировать службы (Services) и проверять работу шаблонов в интерактивном режиме.

Система обладает встроенным инструментарием для архивирования и анализа данных. Компоненты «История» (History) и «Журнал событий» (Logbook) позволяют визуализировать динамику изменения переменных в виде графиков и списков событий. Данные аккумулируются в базе данных (SQLite по умолчанию или внешние SQL-базы) и могут быть использованы для анализа эффективности работы систем и расследования инцидентов.

Для того чтобы шлюзы и модули RealLab могли работать в среде Home Assistant, на аппаратную платформу устанавливается операционная система (Home Assistant OS) или среда исполнения в контейнере. Изготовитель оборудования обеспечивает преднастройку базовых компонентов и поддержку драйверов (интеграций) для своих модулей, что позволяет конечному пользователю сразу перейти к разработке прикладной логики.

Home Assistant включает в себя гибкую систему визуализации (Lovelace Dashboards), которая используется для операторского управления. Интерфейс адаптивен и может быть запущен в любом современном веб-браузере, на мобильных устройствах или специализированных панелях управления. Пользователь может создавать неограниченное количество панелей с различным уровнем доступа и графическим представлением.

Пользователь может самостоятельно расширять возможности системы путем установки дополнительных компонентов (Add-ons) и кастомных интеграций через сообщество. Это позволяет добавлять поддержку нестандартных протоколов или специфических облачных сервисов.

Для обеспечения надежности Home Assistant поддерживает механизмы резервного копирования и восстановления системы. При внезапном отключении питания система сохраняет последние состояния сущностей и накопленные данные в энергонезависимой базе данных, что гарантирует преемственность работы алгоритмов после перезагрузки.

1.3. Преимущества Home Assistant перед другими IoT-платформами

Использование платформы Home Assistant (в частности, в качестве предустановленного ПО на шлюзах серии NLS-Gateway-xx-HA) имеет ряд объективных преимуществ перед другими IoT-решениями так и перед классическими закрытыми промышленными SCADA-системами начального уровня.

Ключевыми преимуществами платформы являются:

- **Локальное выполнение и автономность (Local-first).** В отличие от коммерческих облачных платформ, где обработка сценариев происходит на удаленных серверах, Home Assistant работает полностью локально на мощностях шлюза. Это обеспечивает:
 - надежность: все сценарии автоматизации, опросы проводных и беспроводных модулей и логика управления выполняются непосредственно на шлюзе. Отключение интернета или блокировка внешних облачных сервисов не влияют на работоспособность системы;
 - безопасность данных: история показаний датчиков, видеопотоки и параметры технологических процессов хранятся локально на накопителе шлюза и не передаются на сторонние серверы без явного желания пользователя.
- **Широкая совместимость и поддержка стандартов.** Home Assistant является одним из наиболее универсальных средством автоматизации в сфере благодаря поддержке более чем 2500 официальных интеграций:
 - объединение сред: платформа позволяет в рамках одного шлюза связать промышленные проводные стандарты (Modbus RTU, CAN) с беспроводными протоколами автоматизации зданий (Zigbee, LoRa, Wi-Fi, Bluetooth) и сетевыми протоколами (MQTT, HTTP, SNMP).
 - гибкость выбора: пользователь может сочетать промышленные модули ввода-вывода серии NLS для ответственных узлов с доступными беспроводными датчиками сторонних производителей для вспомогательных задач.
- **Отсутствие лицензионных ограничений и зависимости от вендора.** За счет этого обеспечивается:

- экономическая эффективность: будучи программным обеспечением с открытым исходным кодом, Home Assistant не требует покупки лицензий.
- защита от санкций и закрытия сервисов: пользователь не зависит от политики обновлений конкретного зарубежного бренда или стабильности его облачной инфраструктуры.
- **Масштабируемый и гибкий движок автоматизации.** Платформа предоставляет инструменты разработки логики под задачи разной сложности:
 - для простых связей используется встроенный визуальный конструктор;
 - сложные алгоритмы с ветвлением, циклами и математическими расчетами могут быть описаны в текстовом формате YAML;
 - при необходимости глубокой аналитики или использования редких протоколов связи возможна интеграция со средой визуального программирования Node-RED.
- **Готовая экосистема визуализации и уведомлений.** За счет этого обеспечивается:
 - кроссплатформенность: мобильное приложение Home Assistant Companion (для Android и iOS) позволяет без дополнительной разработки интерфейса управлять системой со смартфона, использовать функции геолокации и получать push-уведомления о нештатных ситуациях на объекте;
 - кастомизация панелей: графический интерфейс Lovelace позволяет быстро создавать панели оператора.

1.4. Шлюзы Reallab поддерживающие Home Assistant

1.4.1. Модельный ряд промышленных IoT шлюзов Reallab

Шлюзы серии NLS-Gateway выпускаются в нескольких модификациях, различающихся набором интерфейсов и предустановленным программным обеспечением. Ниже приведён перечень моделей, поставляемых со средой визуального программирования Home Assistant (суффикс -HA):

- NLS-Gateway-LoRa-2RS-HA;
- NLS-Gateway-LoRa-2RS- PoE-HA;

- NLS-Gateway-Zigbee-2RS-HA;
- NLS-Gateway-Zigbee-2RS-PoE-HA;

Примечание: кроме перечисленных моделей, есть модификации с суффиксами -CDS (CODESYS) и -NR (Node-RED).

Шлюз представляет собой многофункциональное устройство, предназначенное для построения распределённых систем сбора данных и управления на базе интерфейсов RS-485 (Modbus RTU), беспроводных каналов LoRaWAN, Zigbee и Wi-Fi. Шлюз работает под управлением встроенной операционной системы «RealLab! Raspbian Linux arm64», что обеспечивает надёжность серверного класса в жёстких условиях эксплуатации (промышленные температуры, вибрации, электромагнитные помехи).

1.4.2. Основные характеристики шлюза

- **Процессор:** Broadcom BCM2711 (4 ядра ARM Cortex-A72, тактовая частота 1.5 ГГц).
- **Операционная система:** RealLab! Raspbian Linux arm64.
- **Оперативная память (ОЗУ):** 2 ГБ.
- **Системная память:** 16 ГБ eMMC (встроенная флеш-память).
- **Интерфейсы:** UART, I²C, SPI, CAN, Ethernet, USB, LoRa, Wi-Fi, Bluetooth.
- **Питание:** 10–30 В постоянного тока (диапазон может варьироваться в зависимости от модели).
- **Рабочая температура:** от –20 °C до +70 °C.
- **Крепление:** DIN-рейка.

1.4.3. Программное обеспечение и поддерживаемые протоколы

Все шлюзы серии NLS-Gateway работают под управлением ОС RealLab! Raspbian Linux arm64. Состав предустановленного программного обеспечения и набор поддерживаемых протоколов зависят от модификации шлюза.

1.4.3.1. Программное обеспечение и средства разработки

Шлюзы предоставляют следующие программные средства для разработки и настройки:

- **Веб-конфигуратор RealLab** – для начальной настройки шлюза и сетевых параметров.

- **Qt Creator** – среда разработки для создания системных приложений на C++ и QML.
- **Языки программирования:** C/C++ (системное и прикладное ПО), Python (скрипты и автоматизация), JavaScript (веб-интерфейсы и пользовательские приложения).

В зависимости от суффикса в конце наименования модели шлюз поставляется с одной из следующих сред разработки:

- «**NR**» – предустановлена среда Node-RED (визуальное программирование для интеграции устройств и построения логики обработки данных).
- «**CDS**» – предустановлена среда CODESYS (программирование по стандарту МЭК 61131-3 для задач промышленной автоматизации).
- «**HA**» – предустановлена среда Home Assistant (платформа для создания систем умного дома и мониторинга).

Поддерживаемые беспроводные протоколы.

Выбор беспроводного протокола определяется установленным модулем:

- Модели с индексом «**LoRa**» (например, NLS-Gateway-LoRa-...) поддерживают LoRa / LoRaWAN (диапазон 864–870 МГц, до 8 каналов приёма).
- Модели с индексом «**Zigbee**» (например, NLS-Gateway-Zigbee-...) поддерживают Zigbee 3.0 (стек Z-Stack 3.x, диапазон 2405–2480 МГц).
- Все шлюзы также имеют встроенный **Wi-Fi** (IEEE 802.11 b/g/n/ac) и Bluetooth 5.0 BLE, которые работают независимо от основного беспроводного модуля.

1.4.3.2. Поддерживаемые промышленные протоколы

Все модели поддерживают:

- Modbus RTU – через порты RS-485;
- Modbus TCP – через порт Ethernet;
- MQTT – для обмена данными с облачными и локальными IoT-платформами (встроенный MQTT-брокер).

Модели с интерфейсом CAN (в названии присутствует «lRS-ICAN») дополнительно поддерживают протокол CANOpen.

Таким образом, состав ПО определяется суффиксом (-NR/-CDS/-HA), тип беспроводного протокола – индексом (LoRa/Zigbee), а поддержка CANOpen – наличием CAN-интерфейса. Все остальные протоколы (Modbus, MQTT) и базовые средства разработки являются общими для всей серии шлюзов.

1.4.4. Интерфейс шлюзов

Шлюзы серии NLS-Gateway оснащены широким набором проводных и беспроводных интерфейсов, обеспечивающих гибкость подключения к различным промышленным сетям и устройствам. Состав интерфейсов может незначительно варьироваться в зависимости от модели (наличие CAN, количество портов RS-485, тип беспроводного модуля). Перечень основных интерфейсов представлен в табл. 1.

Табл. 1. Интерфейсы шлюзов серии NLS-Gateway

Интерфейс	Описание	Примечание
Ethernet	Один порт 10/100/1000Base-T (RJ-45)	Для подключения к локальной сети, настройки, обмена данными по Modbus TCP, MQTT и другим протоколам.
RS-485	Количество портов — от 1 до 2. Обозначение в операционной системе: /dev/com1, /dev/com2.	Для подключения устройств по протоколу Modbus RTU. Порты имеют гальваническую изоляцию и защиту от перенапряжения.
CAN	1 порт (в моделях с опцией 1RS-1CAN)	Для подключения устройств по протоколу CANOpen.
USB 2.0	1 порт Type-A	Для подключения внешних устройств (флеш-накопители, адаптеры, ключи защиты).
Micro-SD	Слот для карт памяти	Поддержка карт microSD объёмом до 128 Гбайт. Используется для расширения системной памяти,

		хранения резервных копий и журналов.
SMA	2 разъёма	Для подключения внешних антенн беспроводных модулей (LoRa/Zigbee и Wi-Fi).
LoRa / LoRaWAN	Встроенный модуль (для LoRa-моделей)	Диапазон частот 864–870 МГц, до 8 каналов приёма, мощность до +27 дБм.
Zigbee 3.0	Встроенный модуль (для Zigbee-моделей)	Диапазон частот 2405–2480 МГц, 16 каналов, мощность до +20 дБм.
Wi-Fi	Встроенный модуль IEEE 802.11 b/g/n/ac	Поддержка режимов клиента и точки доступа.
Bluetooth	Bluetooth 5.0 BLE	Для подключения периферийных устройств и настройки.
Часы реального времени (RTC)	Встроенные	С автономным питанием, для временных метки событий.

1.4.5. Сетевые интерфейсы и настройка связи: Настройка Ethernet и Wi-Fi (Клиент / Точка доступа)

Шлюзы серии NLS-Gateway имеют два независимых сетевых интерфейса, работающих одновременно:

- **Ethernet (eth0)** – проводной интерфейс для подключения к локальной сети.
- **Wi-Fi (wlan0)** – беспроводной интерфейс, который может работать в одном из двух режимов: клиент (подключение к существующей Wi-Fi сети) или точка доступа (создание собственной сети для локального доступа).

1.4.5.1. Настройка Ethernet

По умолчанию интерфейс Ethernet настроен на получение IP адреса по протоколу DHCP. При подключении к сети с DHCP-сервером шлюз автоматически получает динамический IP адрес.

Для настройки **статического IP адреса** через веб-конфигуратор RealLab, подробное смотрите пункт 2.10 «**Настройка параметров сети**» в руководстве пользователя по работе с ПО шлюзов RealLab! серии NLS-Gateway, руководство можно скачать на странице шлюза на сайте компании RealLab.

1.4.5.2. Настройка Wi-Fi

Режим «Точка доступа» (Access Point). Это позволяет подключаться к шлюзу напрямую без наличия внешней сети.

- **Имя сети (SSID):** соответствует имени хоста (Hostname) шлюза, указанному на корпусе (например, NLS-Gateway-LoRa-XXXX).
- **Пароль:** 12345678.
- **IP-адрес шлюза в этом режиме:** 10.10.0.1 (статический).

Для изменения параметров точки доступа (SSID, пароль, канал) используется веб-конфигуратор шлюза. подробное смотрите пункт 2.10.2 «Настройка Wi-Fi» в руководстве пользователя по работе с ПО шлюзов RealLab! серии NLS-Gateway.

Режим «Клиент» (Client). По умолчанию шлюз настроен как Клиент. В этом режиме шлюз подключается к существующей Wi-Fi сети. Для настройки:

1. Подключитесь к веб-интерфейсу шлюза.
2. Перейдите в раздел настройки сети («Wireless & Wired Network Settings»).
3. Выберите беспроводной интерфейс (wlan0).
4. Укажите режим «Клиент».
5. Введите SSID и пароль целевой Wi-Fi сети.
6. Настройте получение IP-адреса (DHCP или статический).
7. Примените настройки. Шлюз переключится в режим клиента и подключится к указанной сети.

Подробное информация по настройке смотрите в [руководстве пользователя по работе с ПО шлюзов RealLab! серии NLS-Gateway](#) в пункте «Настройки IPv4 для Wi-Fi».

ВАЖНО! При переключении в режим клиента доступ к шлюзу по-старому IP-адресу точки доступа (10.10.0.1) может быть потерян. Для дальнейшего доступа используйте IP-адрес, полученный от внешней сети.

1.4.5.3. Одновременная работа интерфейсов

Интерфейсы Ethernet и Wi-Fi работают параллельно и не мешают друг другу. Это позволяет организовать резервирование каналов связи или разделить потоки данных (например, Ethernet – для промышленных протоколов, Wi-Fi – для удалённого мониторинга и настройки).

2. Начало работы

На шлюзах Reallab серии NLS-Gateway-xx-NA ПО платформа Home Assistant предустановлена по умолчанию.

2.1. Подключение к шлюзу и запуск интерфейса Home Assistant

Для начала работы со шлюзом серии NLS-Gateway под управлением IoT платформы Home Assistant необходимо:

1. Закрепите шлюз NLS-Gateway на DIN-рейке в месте, обеспечивающем оптимальный радиоприем.
2. Далее установите внешнюю антенну LoRaWAN на ВЧ-разъем типа SMA шлюза, а также антенну Wi-Fi (при использовании беспроводной сети).
3. Подключите источник питания к контактам +Vs и GND съёмной клеммной колодки. После включения загорится индикатор «Работа».
4. Подключите кабель Ethernet от локальной сети к сетевому разъёму шлюза. Шлюз по умолчанию получает IP-адрес по протоколу DHCP. Чтобы узнать текущий IP-адрес шлюза, воспользуйтесь утилитой RealLab! Scanner v2.0 (доступна на сайте производителя).
5. В верхней части окна утилиты выберите сетевые интерфейсы, через которые будет выполняться поиск, и при необходимости добавьте подсеть (например, 192.168.0.1) (Рис. 2.1).

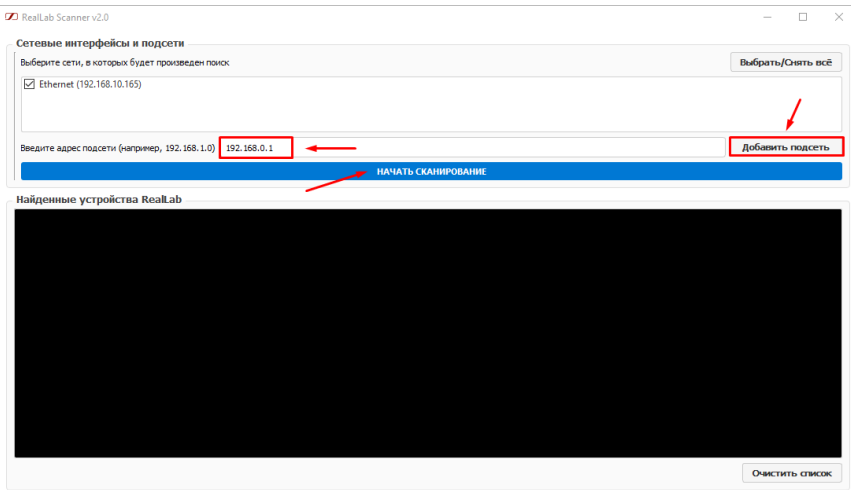


Рис. 2.1. Общий вид утилиты RealLab! Scanner v2.0

6. Нажать «Начать сканирование». Найденные устройства отобразятся в формате: [IP-адрес] → Имя хоста (Hostname) (рис. 2.2).

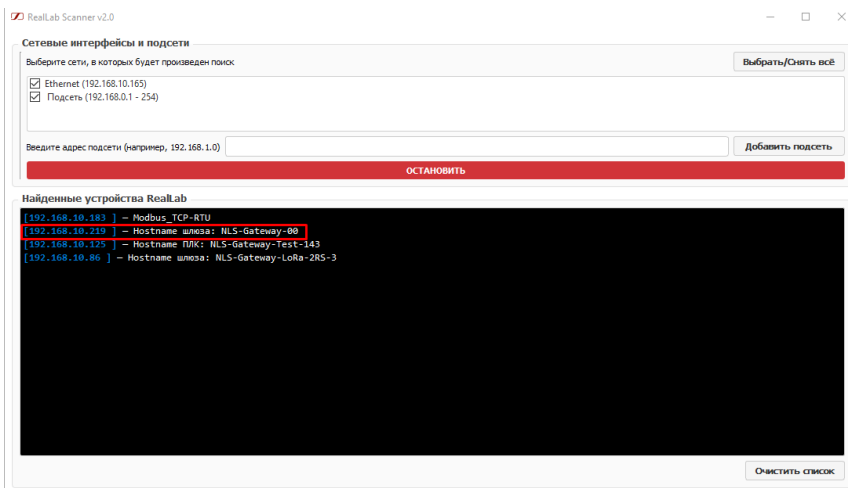


Рис. 2.2. Найденные устройства в окне результатов

2.2. Доступ к веб-интерфейсу Home Assistant

Для открытия веб-интерфейса Home Assistant необходимо открыть браузер на ПК, находящимся в одной сети со шлюзом, и через адресную строку открыть порт 8123 на IP-адресе шлюза, например: <http://192.168.10.86:8123/>

Также открыть веб-интерфейс Home Assistant можно из интерфейса веб-конфигуратора шлюза. Для открытия веб-конфигуратора необходимо открыть в браузере http://IP_шлюза:8000/.

2.2.1. Авторизация в Home Assistant

При первом запуске Home Assistant предложит «Создать мой умный дом». Нажимаем кнопку (рис. 2.3).

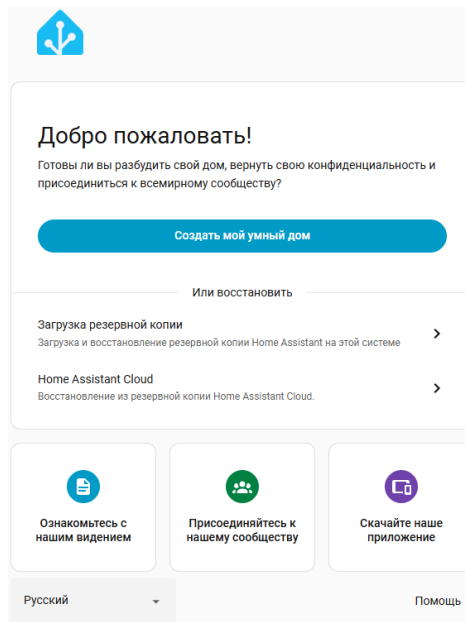


Рис. 2.3. Стартовая страница создания умного дома

Затем, заполняя все поля, создаём пользователя (рис. 2.4).

Создать пользователя

Давайте начнем с создания учётной записи пользователя.

Имя*

Имя пользователя*

Пароль*

Выберите надежный и уникальный пароль. Обязательно сохраните его, чтобы не забыть.

Подтвердите пароль*

Создать учётную запись

Русский

Помощь

Рис. 2.4. Страница создания пользователя

Затем выбираем любое местоположения. Выбранное местоположение не влияет на работоспособность. (рис. 2.5).

Местоположение дома

Эти данные хранятся только в вашей системе Home Assistant, хотя некоторые облачные интеграции могут использовать их для своей работы.

Поиск адреса

AMSTERDAM

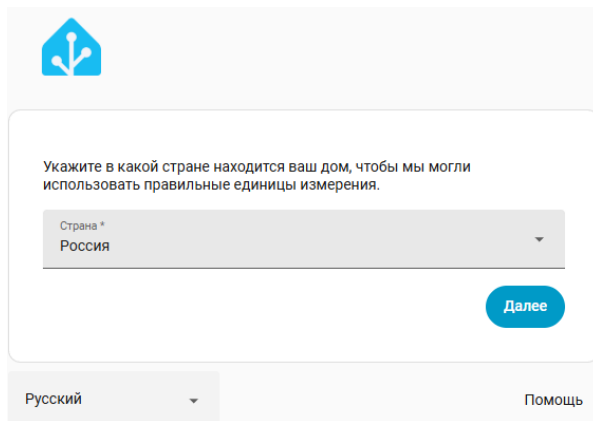
Далее

Русский

Помощь

Рис. 2.5. Страница выбора местоположения

Затем выбираем страну. (рис. 2.6).



Укажите в какой стране находится ваш дом, чтобы мы могли использовать правильные единицы измерения.

Страна *
Россия

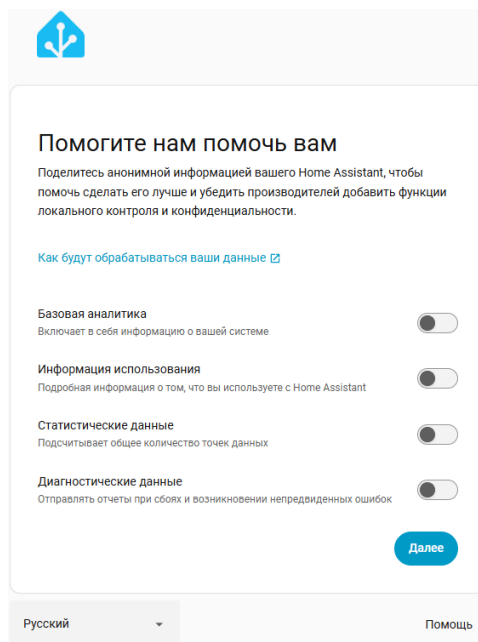
Далее

Русский

Помощь

Рис. 2.6. Страница выбора страны

Затем нажимаем «далее», ничего не выбирая. (рис. 2.7).



Помогите нам помочь вам

Поделитесь анонимной информацией вашего Home Assistant, чтобы помочь сделать его лучше и убедить производителей добавить функции локального контроля и конфиденциальности.

[Как будут обрабатываться ваши данные](#)

Базовая аналитика
Включает в себя информацию о вашей системе

Информация использования
Подробная информация о том, что вы используете с Home Assistant

Статистические данные
Подсчитывает общее количество точек данных

Диагностические данные
Отправлять отчеты при сбоях и возникновении непредвиденных ошибок

Далее

Русский

Помощь

Рис. 2.7. Страница разрешения на обработку данных

Завершаем авторизацию, нажимая «Готово». (рис. 2.8). Откроется начальный экран Home assistant (рис. 2.9).

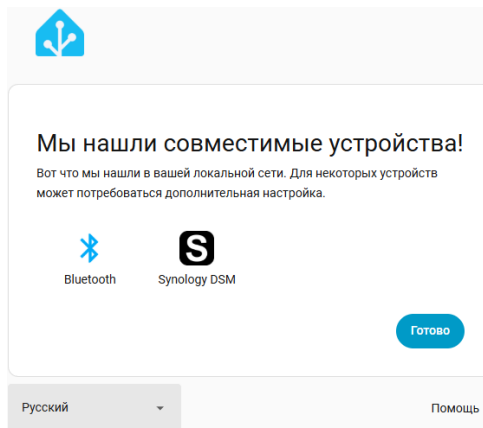


Рис. 2.8. Страница найденных устройств в локальной сети

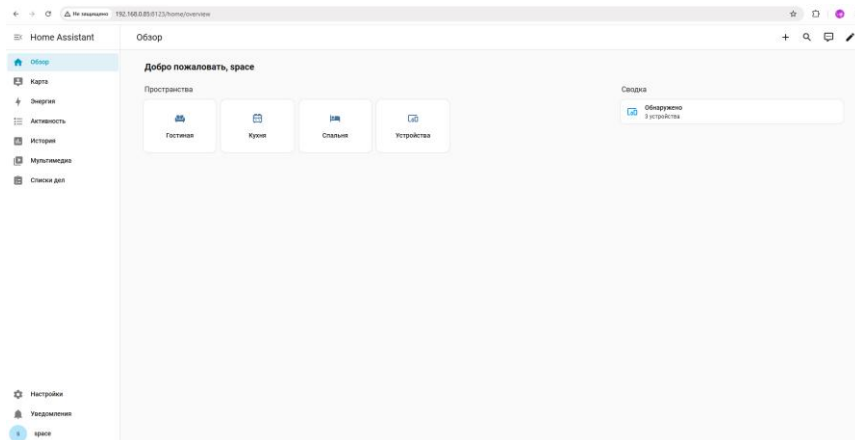


Рис. 2.9. Окно Home Assistant

После авторизации Home Assistant готов к работе.

Для создания скриптов, панелей вывода сначала нужно добавить устройства ввода-вывода. Ниже представлена инструкция по подключению устройств, работающих на различных протоколах.

ВАЖНО! При первом запуске Home Assistant (на этапе инициализации системы) создается единственная учетная запись с ролью Владельца (Owner). Ее создание происходит только один раз, и к сохранности этих параметров (логина и пароля) следует отнестись ответственно.

Рекомендация:

Сразу после завершения первоначальной настройки сохраните учетные данные Владельца в надежном месте — например, в менеджере паролей.

2.3. Работа с интерфейсом Home Assistant

Интерфейс Home Assistant является основным инструментом для мониторинга, ручного управления и конфигурирования системы автоматизации на шлюзах серии NLS-Gateway-NA. Графическая оболочка адаптивна и оптимизирована для работы как на персональных компьютерах, так и на мобильных устройствах.

Интерфейс разделен на три ключевые зоны:

- **Боковое меню навигации** (слева) — служит для переключения между функциональными экранами.
- **Верхняя панель управления** (справа вверху) — отображает контекстные действия текущего экрана (поиск, настройки отображения, вызов дополнительных меню).
- **Основная рабочая область** (справа по центру) — область вывода карточек управления, таблиц конфигурации, логов и графиков.

2.3.1. Элементы боковой панели навигации

Боковая панель предоставляет доступ к ключевым подсистемам Home Assistant. Перечень доступных пунктов может расширяться по мере установки дополнительных интеграций и дополнений (Add-ons). Стандартный набор включает в себя следующие разделы:

- **Обзор (Overview):** главная панель мониторинга и управления (Dashboard), построенная на базе интерфейса *Lovelace*. По умолчанию Home Assistant автоматически группирует здесь все обнаруженные устройства и датчики. Пользователь может создавать собственные специализированные панели для различных задач или экранов.

- **Карта (Map):** интерактивная географическая карта, на которой отображаются зоны (например, «Дом», «Офис») и местоположение мобильных устройств с установленным приложением Home Assistant Companion.
- **История (History):** экран ретроспективного анализа. Позволяет выбрать произвольный период времени, группы сущностей или конкретные датчики и построить графики изменения их состояний.
- **Журнал (Logbook):** детализированная хронологическая лента событий. Фиксирует точное время изменения состояний сущностей (например, «Реле 1 включено автоматизацией», «Сработал датчик движения в 12:01:15»).
- **Энергия (Energy):** специализированная панель для мониторинга энергопотребления, выработки электроэнергии (например, от солнечных панелей) и потребления воды или газа. Требуется привязки датчиков со специальным классом устройства (`device_class: energy`).
- **Инструменты разработчика (Developer Tools):** сервисный раздел, необходимый для отладки конфигурации, тестирования скриптов, ручного изменения состояний сущностей и прослушивания внутренних системных событий.
- **Настройки (Settings):** главная конфигурационная панель, через которую осуществляется администрирование шлюза, подключение оборудования, настройка сценариев автоматизации и обновление ПО.

2.3.2. Элементы боковой панели навигации

В самом низу боковой панели отображается аватар и имя текущего пользователя. Нажатие на эту иконку открывает экран индивидуальных настроек пользователя:

- **Язык (Language) и Регион:** настройка локализации интерфейса, формата времени и числовых разделителей.
- **Тема оформления (Theme):** переключение между светлой, темной или системной темами оформления, а также выбор кастомных тем (если установлены).
- **Режим изменения порядка меню:** позволяет скрыть неиспользуемые разделы боковой панели или изменить последовательность их отображения.

2.3.3. Подробный разбор раздела «Настройки» пользователя

Раздел «Настройки» является административным ядром шлюза. Он разделен на несколько разделов:

Раздел «Устройства и службы» - основная рабочая область для интеграции аппаратного обеспечения. Раздел содержит четыре внутренних подраздела:

- **Интеграции (Integrations):** отображает активные драйверы протоколов и сервисов (например, Modbus, MQTT, Zigbee Home Automation). Кнопка «Добавить интеграцию» в правом нижнем углу запускает мастер поиска и добавления новых протоколов взаимодействия.
- **Устройства (Devices):** физические или виртуальные приборы (например, модуль *NLS-16DI*). При нажатии на конкретное устройство открывается его индивидуальная карточка, где сгруппированы все связанные с ним датчики и управляющие элементы, автоматизации, в которых оно задействовано, и диагностические параметры.
- **Сущности (Entities):** полный плоский список всех системных переменных. Здесь можно осуществлять массовый поиск, временно отключать неиспользуемые сущности, изменять их системные имена (Entity ID) или группировать их.
- **Вспомогательные (Helpers):** вкладка для создания виртуальных сущностей. Вспомогательные объекты используются для хранения промежуточных данных или управления логикой без физического оборудования. Доступны следующие типы помощников:
 - **Входной переключатель (input_boolean):** виртуальный тумблер (Вкл/Выкл).
 - **Входное число (input_number):** ползунок или поле ввода для хранения числового значения (например, уставка температуры).
 - **Входной текст (input_text):** текстовое поле.
 - **Таймер (timer):** программный таймер обратного отсчета для сценариев автоматизации.
 - **Шаблон (template):** виртуальный сенсор, значение которого рассчитывается по формуле (например, разница показаний двух датчиков температуры).

Раздел «Автоматизации и сцены» - раздел создания логических связей и алгоритмов работы «умного дома». Раздел разбит на 4 подраздела:

- **Автоматизации (Automations):** алгоритмы, работающие по принципу «триггер — условие — действие».
 - **Триггеры (Triggers):** события, запускающие проверку правила (например, «время стало 18:00» или «значение датчика превысило порог»).
 - **Условия (Conditions):** дополнительные фильтры (например, «выполнять действие только если наступила ночь»).
 - **Действия (Actions):** конечные команды, отправляемые на исполнительные механизмы (например, «включить реле №3»).
- **Сцены (Scenes):** зафиксированные статические состояния группы устройств (например, сцена «Ушел» переводит все реле в выключенное состояние, а климатическую систему — в режим экономии).
- **Скрипты (Scripts):** жестко заданные последовательности действий, которые можно запускать вручную из интерфейса или вызывать из различных автоматизаций для исключения дублирования кода.
- **Чертежи (Blueprints):** готовые шаблоны автоматизаций и скриптов, созданные сообществом. Позволяют быстро настраивать типовые задачи (например, управление освещением по датчику движения) без необходимости написания логики с нуля.

Раздел «Панель» Раздел управления интерфейсом пользователя. Здесь можно создавать новые панели Lovelace, разграничивать права доступа к ним для разных учетных записей или задавать стартовую панель для конкретного устройства (планшета, смартфона).

Раздел «Система» - низкоуровневое администрирование шлюза Reallab:

- **«Информация о доме»:** настройка географических координат шлюза (необходима для корректного расчета времени восхода и заката солнца), единиц измерения (метрическая/имперская система) и часового пояса.
- **«Обновления»:** просмотр и установка обновлений ядра Home Assistant (Core), операционной системы шлюза (Operating System) и установленных дополнений.
- **«Резервное копирование»:** создание полных или частичных резервных копий конфигурации, баз данных и установленных аддонов. Копии можно скачивать локально на ПК. Крайне рекомендуется создавать бэкап перед любыми изменениями конфигурационных файлов или обновлением системы.

- **«Сеть»:** настройка сетевых адаптеров шлюза (Ethernet, Wi-Fi), выбор режима получения IP-адреса (DHCP или статический IP), настройка прокси-серверов и сетевых протоколов (IPv4/IPv6).
- **«Журнал сервера»:** важнейший инструмент диагностики. Сюда выводятся системные ошибки, предупреждения интеграций и отчеты о сбоях связи (например, таймауты при опросе устройств по Modbus TCP/RTU).
- **«Перезапуск Home Assistant»** справа сверху меню. Содержит кнопки «Быстрый перезапуск» (перезапуск только программной оболочки) и «Перезапуск Home Assistant» (полная перезагрузка операционной системы шлюза).

2.3.4. Инструменты разработчика

Данный раздел предназначен для пусконаладочных работ и глубокого анализа поведения системы. Он состоит из пяти основных вкладок:

- **Состояния (States):**

Таблица всех зарегистрированных в системе сущностей. Для каждой сущности отображаются три параметра:

- **Entity ID:** уникальный системный идентификатор (например, `sensor.temperature_sensor_1`).
- **State (Состояние):** текущее значение переменной (например, 24.5 или on). Пользователь может вручную ввести любое значение в этом поле для имитации срабатывания датчика при тестировании автоматизаций.
- **Attributes (Атрибуты):** метаданные сущности в формате JSON (например, единица измерения `unit_of_measurement: °C`, класс устройства `device_class: temperature`, или имя для отображения `friendly_name: Температура на улице`).

- **Действия (Actions):**

Интерфейс для отправки ручных команд в шину Home Assistant. Например, выбрав действие `switch.turn_on` и указав в качестве цели конкретное реле, можно принудительно замкнуть сухой контакт. Здесь же тестируются команды отправки данных в регистры Modbus или публикации топиков в MQTT.

- **Шаблоны (Templates):**

Интерактивный интерпретатор шаблонов на языке *Jinja2*. Используется для динамической обработки данных. Например, выражение `{{ states('sensor.voltage') | float * 2.2 }}` позволяет на лету проверить правильность математических преобразований, которые планируется внедрить в конфигурационные файлы.

- **События (Events):**

Позволяет подписаться на внутреннюю шину событий Home Assistant и в реальном времени отслеживать, какие системные сообщения генерируются устройствами (например, нажатия кнопок на Zigbee-пультах, системные ошибки и т.д.).

2.3.5. Структура сущностей (Entities) в Home Assistant

Все данные, циркулирующие в шлюзе, представляются в виде сущностей (Entities). Любая сущность имеет строго регламентированный формат идентификатора: `Domain.Object_ID`. Где `Domain` определяет тип сущности и набор доступных для неё действий, а `Object_ID` – уникальное имя конкретного физического или виртуального датчика/выключателя.

Основные домены, используемые при работе со шлюзами Reallab:

- **Sensor** - аналоговый датчик только для чтения (температура, влажность, напряжение, ток);
- **binary_sensor** - бинарный (двоичный) датчик только для чтения. Принимает исключительно два состояния: `on` (включено/активно) и `off` (выключено/неактивно). Применяется для дискретных входов типа «сухой контакт», датчиков протечки, движения или открытия дверей;
- **switch** - управляемый двухпозиционный переключатель (чтение и запись). Применяется для управления релейными выходами модулей Reallab;
- **light** - диммируемый или многоцветный источник света. Имеет дополнительные атрибуты яркости и цветовой температуры;
- **number** - числовое поле (ползунок или окно ввода) для записи значений в подключенные приборы (например, задание уставки давления или коэффициента калибровки);
- **select** - выпадающий список для выбора одного из текстовых режимов работы устройства.

3. Подключение устройств. Работа с протоколами

3.1. MQTT

MQTT – лёгкий протокол «публикация/подписка» для IoT. Устройства подключаются к брокеру по TCP (обычно с TLS), публикуют сообщения в темы, другие клиенты на эти темы подписываются.

Роли в MQTT:

- **Publisher** - отправляет сообщения.
- **Subscriber** – получает сообщения.
- **Broker** - маршрутизирует сообщения между ними.

Работа MQTT будет описана на примере модулей ввода-вывода Reallab – NLS-4AO-Ethernet и NLS-4RTD-Ethernet.

3.1.1. Подключение к MQTT брокеру

На шлюзе Reallab предустановлен MQTT брокер Mosquitto. Подключимся к нему через Home Assistant. Для этого перейдём в настройки -> устройства и службы (рис. 3.1).

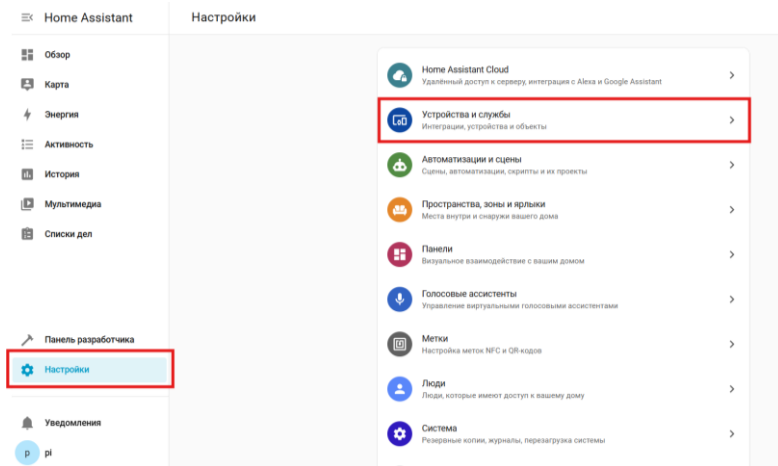


Рис. 3.1. Окно настроек Home Assistant

Подключение устройств. Работа с протоколами

Откроется окно устройств и служб, нажмём на кнопку “Добавить интеграцию” (рис. 3.2).

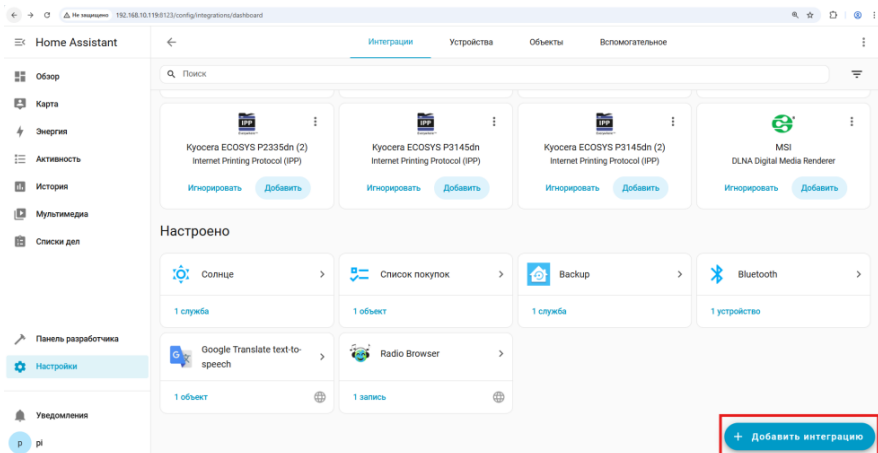


Рис. 3.2. Кнопка “Добавить интеграцию”

Выбираем MQTT, воспользовавшись поиском (рис. 3.3). Затем выбираем снова “MQTT” (рис. 3.4).

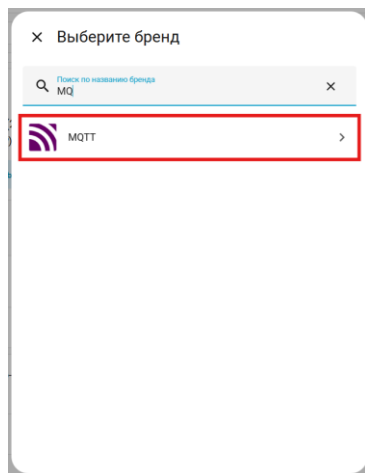


Рис. 3.3. Окно выбора устройства

3. Подключение устройств. Работа с протоколами

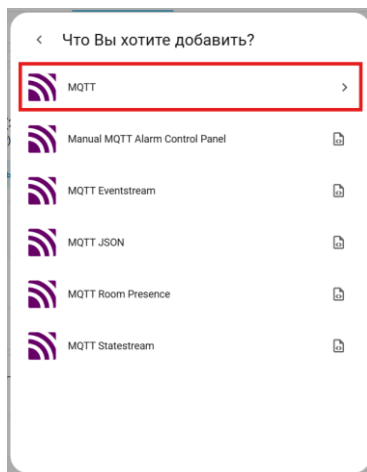


Рис. 3.4. Окно выбора устройства

Вводим настройки брокера (рис. 3.5):

- IP-адрес шлюза, порт = 1883
- Имя пользователя и пароль по умолчанию пустые.

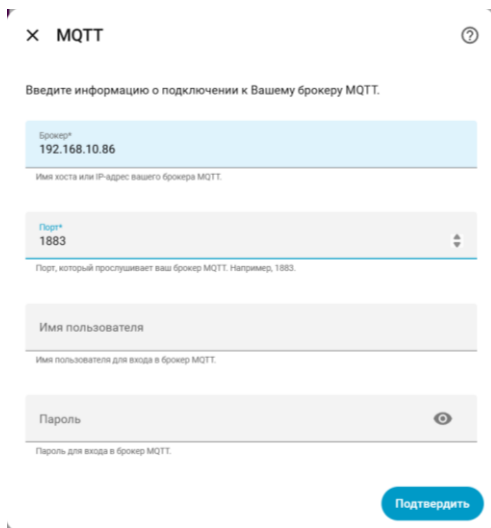


Рис. 3.5. Настройки подключения к MQTT брокеру

Откроется меню службы MQTT, где можно добавлять новые устройства (рис. 3.6).

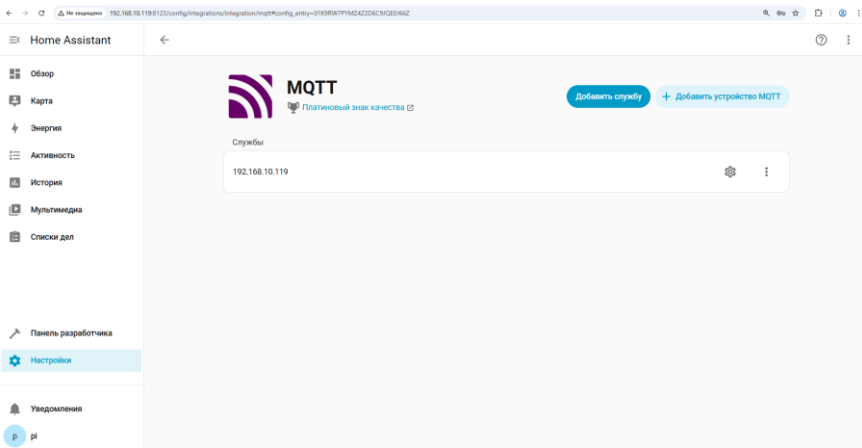


Рис. 3.6. Окно службы MQTT

3.1.2. Публикация команд

1. Подключим модуль аналогового вывода NLS-4AO-Ethernet.
2. Определим топики (адреса) значений для публикации. Для этого воспользуемся программой MQTT Explorer. Подключимся к брокеру и увидим топики модуля (рис. 3.7).

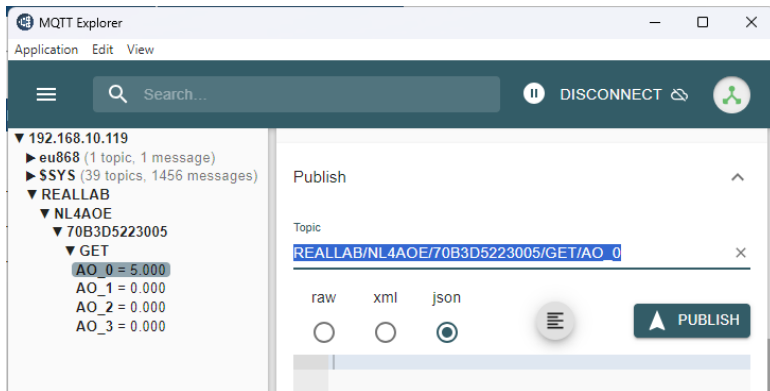


Рис. 3.7. Окно MQTT Explorer

3. Скопируем топик одного из выходов.
4. В Home Assistant нажмём в меню службы “MQTT” кнопку “Добавить устройство MQTT” (рис. 3.8).

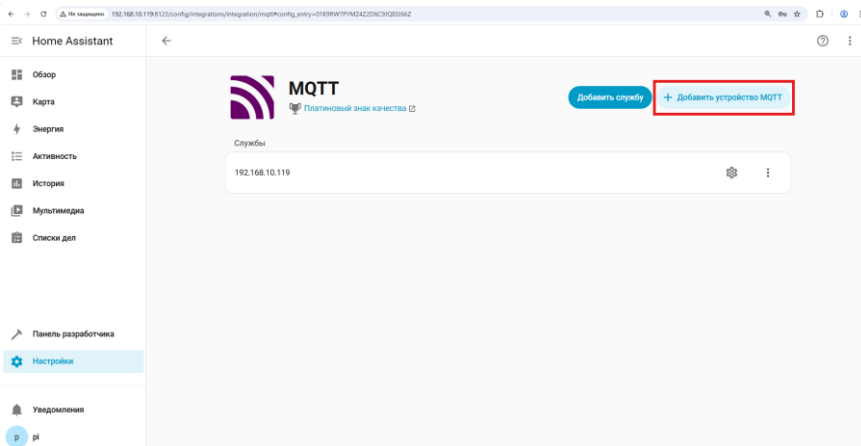


Рис. 3.8. Кнопка “Добавить устройство MQTT”

5. Придумываем и вписываем название и модель устройства (рис. 3.9).

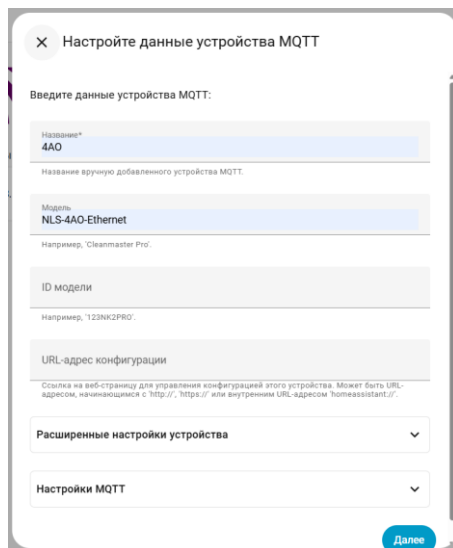


Рис. 3.9. Настройка устройства MQTT

6. Выбираем тип объекта (рис. 3.10).

От типа объекта будет зависеть как устройство выглядит в интерфейсе (график датчика, переключатель, лампа, климат и т.п.), какие действия с ним доступны (только чтение значения, включить/выключить, менять яркость или температуру), как интерпретируется payload из MQTT (число, текст, двоичное состояние, набор параметров) и как эту сущность можно использовать в автоматизациях и статистике.

По сути, выбор типа говорит Home Assistant: “это датчик”, “это выключатель”, “это свет”, “это климат” – и от этого зависят и UI, и команды, и логика работы.

Рис. 3.10. Настройка устройства MQTT

7. Вставляем скопированный ранее топик из MQTT Explorer в поле “Топик команды” (рис. 3.11). На скриншоте представлен тип объекта вентилятора. В этом типе первый топик команды отвечает за вкл/выкл. Мы будем использовать его только для выключения, переводя значение в 0.

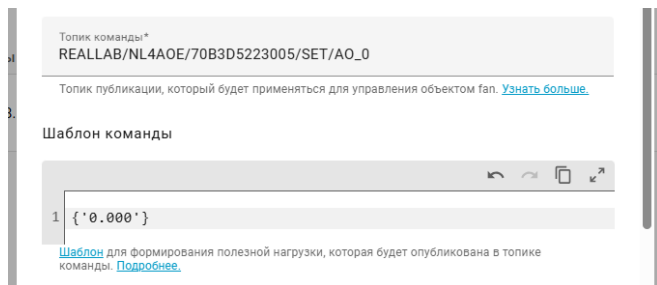


Рис. 3.11. Настройка устройства MQTT

8. Вставляем тот же топик или другой в поле “Топик команды процента” (рис. 3.12). В типе объекта “вентилятор” этот топик отвечает за интенсивность работы вентилятора. Мы будем использовать шаблон команды, что переводить проценты от 0 до 100 в число от 0 до 10 вольт:

```
{{ '{:.3f}'.format((value | float(0)) * 0.1) }}
```

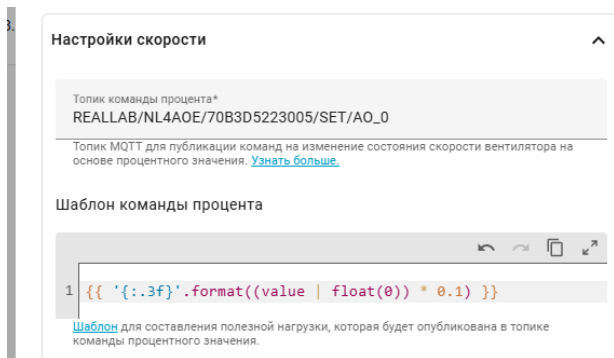


Рис. 3.12. Настройка устройства MQTT

Это последний шаг, после него созданный объект готов к использованию, добавлению на дашборд.

Перезапустите Home Assistant, если объект не появился в списке MQTT.

3.2. Zigbee

Zigbee – энергоэффективный беспроводной протокол ячеистой (mesh) сети для IoT. Устройства связываются по радиоканалу (на базе стандарта IEEE

802.15.4), передавая сообщения напрямую или через соседние узлы для увеличения дальности связи.

Роли в Zigbee:

- **Coordinator (роль Home Assistant)** – запускает сеть и управляет безопасностью.
- **Router** – ретранслирует сообщения и расширяет зону покрытия.
- **End Device** – отправляет данные и работает в спящем (энергосберегающем) режиме.

В Home Assistant существует 2 способа работы с протоколом Zigbee:

- через оригинальную интеграцию Zigbee Home Automation (ZHA);
- через конвертер Zigbee2MQTT.

Каждый способ имеет противоположные другому способу плюсы и минусы (см. табл. 2).

Табл. 2. Сравнение способов работы с Zigbee

Особенность	Zigbee Home Automation (ZHA)	Zigbee2MQTT
Сложность первичного подключения устройств	Легко	Сложнее
Число поддерживаемых устройств	Ограниченное число устройств	Любые устройства Zigbee

С завода на шлюзах Reallab настроен метод подключения через Zigbee2MQTT.

Полный список устройств, поддерживаемых Zigbee Home Automation приведён по ссылке <https://zigbee.blakadder.com/zha.html>.

Далее описана инструкция по использованию каждого метода на примере беспроводного датчика температуры.

3.2.1. Zigbee Home Automation настройка

Сначала нужно отключить Zigbee2MQTT или убедиться, что он отключен.

Управлять состоянием Zigbee2MQTT можно из веб-конфигуратора шлюза.

3. Подключение устройств. Работа с протоколами

1. Открываем веб-конфигуратор по адресу <http://IP-шлюза:8000/> . Авторизуемся. С завода авторизоваться в конфигураторе можно по логину “pi” и паролю “12345678”.
2. Переходим в раздел “Сеть” и во вкладку “Zigbee”, нажимаем на кнопку “Логирование и управление службами” (рис. 3.13).

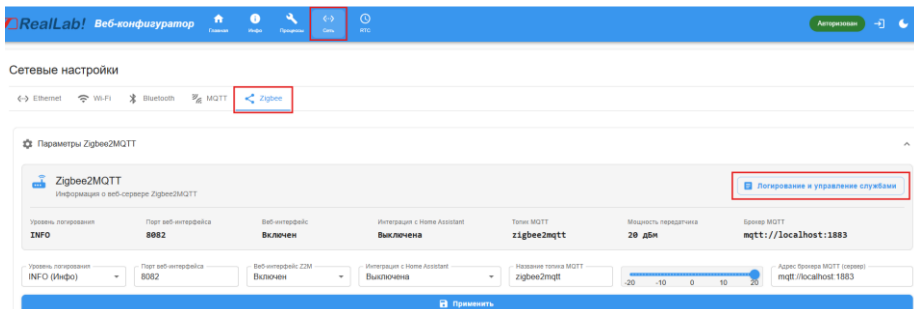


Рис. 3.13. Сравнение способов использования Zigbee

3. В открытом окне нажимаем кнопку “Обновить”.

Если в окне вывода видим “-- No entries --”, сервис Zigbee2MQTT отключен и не мешает работе ZНА, можно приступать к следующему шагу.

Если Zigbee2MQTT активен, отключаем его, нажимая на “остановить” и “убрать из автозапуска” (рис. 3.14). Перезагружаем шлюз через кнопку настроек в правом верхнем углу конфигуратора.

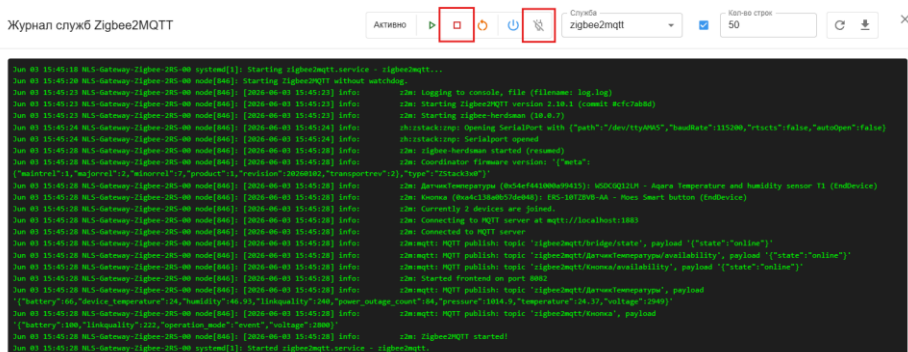


Рис. 3.14. Окно управления службой Zigbee2MQTT

4. Открываем Home Assistant. Переходим в настройки, устройства и службы, добавить интеграцию. Из интеграций выбираем Zigbee Home Automation (рис. 3.15).

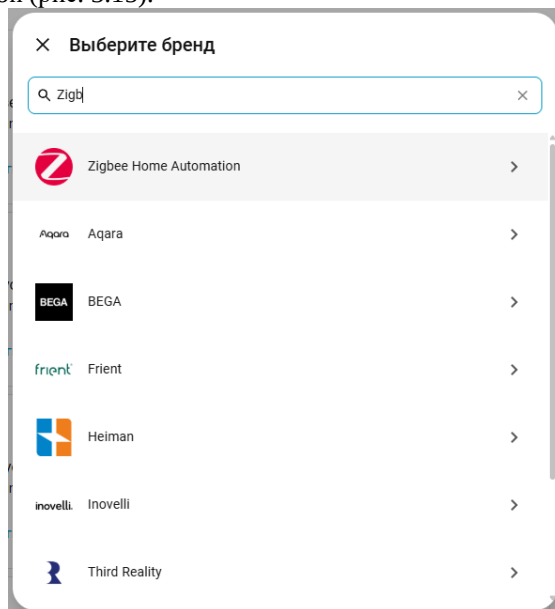


Рис. 3.15. Выбор Zigbee Home Automation из интеграций

5. Нажимаем “Добавить устройство Zigbee” (рис. 3.16).

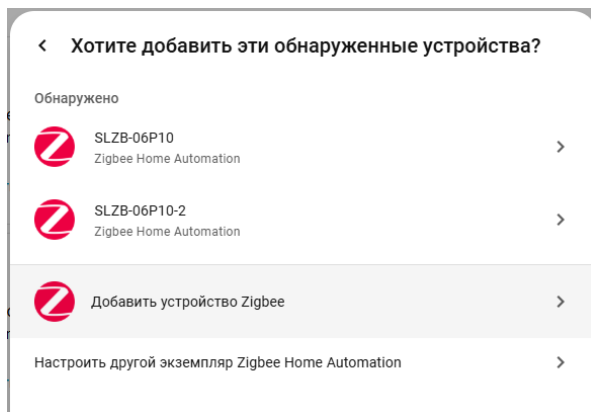


Рис. 3.16. Окно выбора/добавления устройства Zigbee

6. Нажимаем “Продолжить” (рис. 3.17).

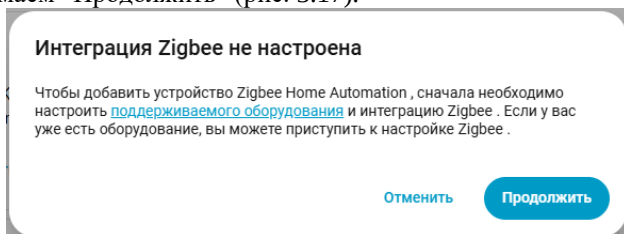


Рис. 3.17. Уведомление о необходимости настроить поддерживаемое оборудование

7. Выбираем ttyAMA5 (рис. 3.18), ожидаем окончания настройки.

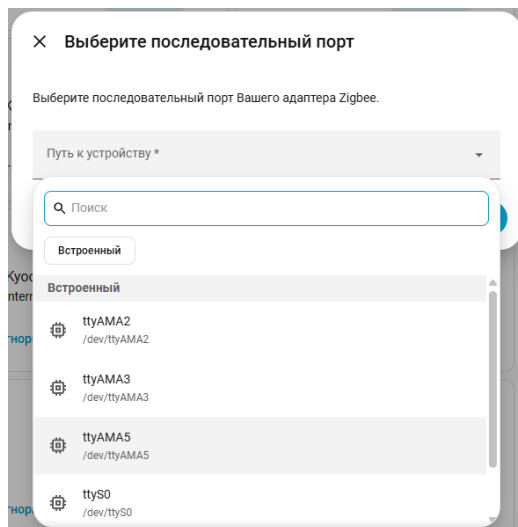


Рис. 3.18. Окно выбора адаптера Zigbee

8. Ожидаем. Нажимаем “Set up automatically” (рис. 3.19).

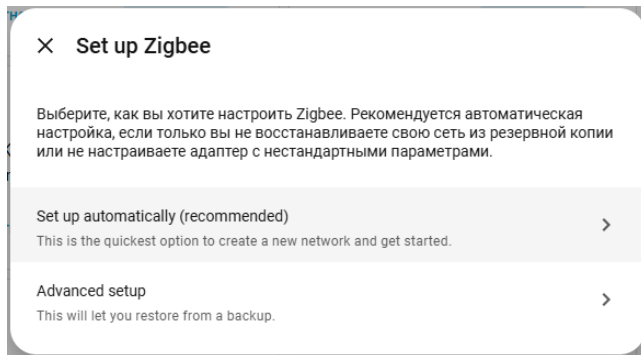


Рис. 3.19. Окно выбора режима установки

9. Ожидаем формирования новой сети (рис. 3.20).

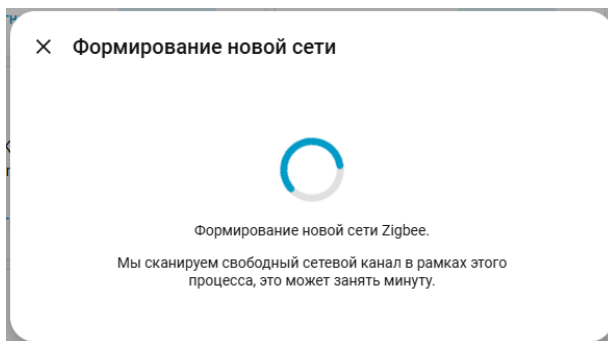


Рис. 3.20. Окно ожидания формирования новой сети

10. Ничего не меняя нажимаем “Готово” (рис. 3.21).

✕ Назвать и назначить



Рис. 3.21. Окно ввода названия адаптера Zigbee

Настройка Zigbee home automation завершена.

3.2.2. Zigbee Home Automation подключение устройств

1. Переходим в настройки, устройства и службы, опускаемся по странице вниз, находим в разделе “Настроено” службу “Zigbee Home Automation”. Открываем её (рис. 3.22).

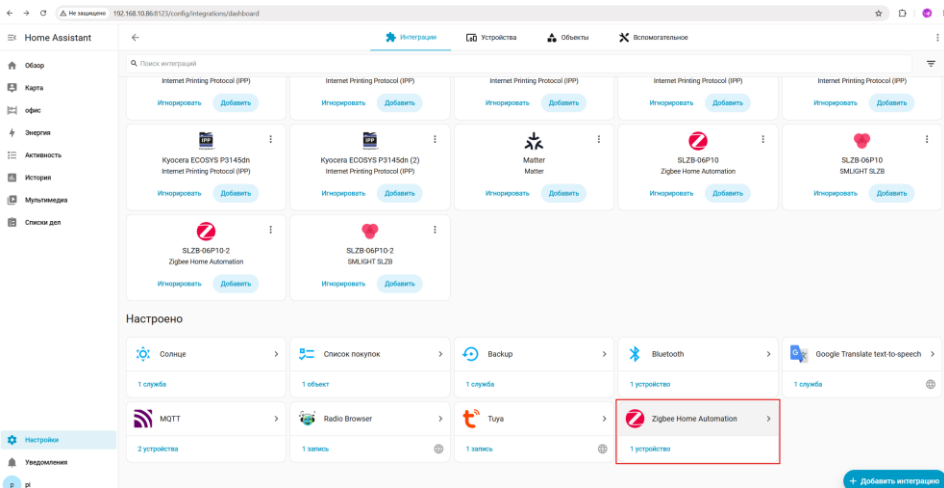


Рис. 3.22. Служба Zigbee Home Automation

2. Нажимаем “Добавить устройство” (рис. 3.23).

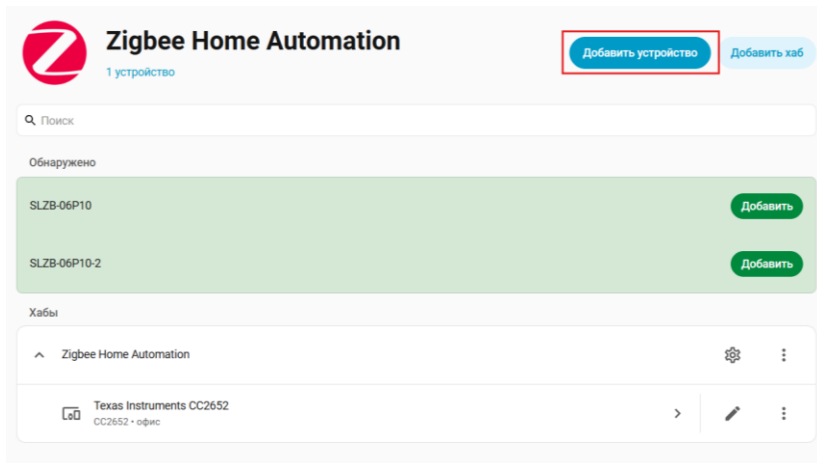


Рис. 3.23. Страница интеграции Zigbee Home Automation

3. Пока происходит поиск Zigbee устройств, переводим подключаемое устройство в состояние подключения. Например, зажимаем кнопку, пока светодиод на устройстве не начнет мигать.

Home Assistant обнаружит устройство (рис. 3.24). Затем home assistant будет ждать первой отправки пакета данных для настройки. Можно подождать или нажать кнопку принудительной отправки на Zigbee устройстве.

После этого настройка будет завершена (рис. 3.25). Можно задать устройству имя и пространство или оставить всё по умолчанию и выйти из поиска Zigbee устройств.

Поиск Zigbee устройств...

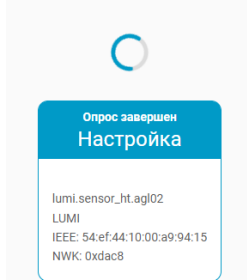


Рис. 3.24. Окно поиска Zigbee устройств с найденным и настраиваемым устройством

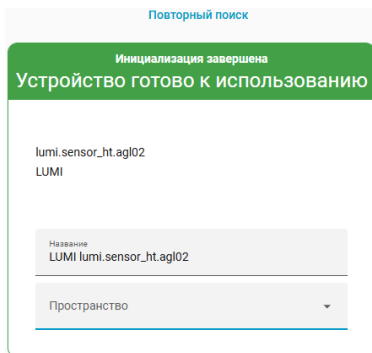


Рис. 3.25. Окно поиска Zigbee устройств с найденным и настроенным устройством

4. Устройство соединено, Home Assistant принимает все передаваемые им пакеты. Подключенное устройство можно найти во вкладке устройств в настройках (рис. 3.26 – рис. 3.27). Его параметры можно добавлять на дашборд.

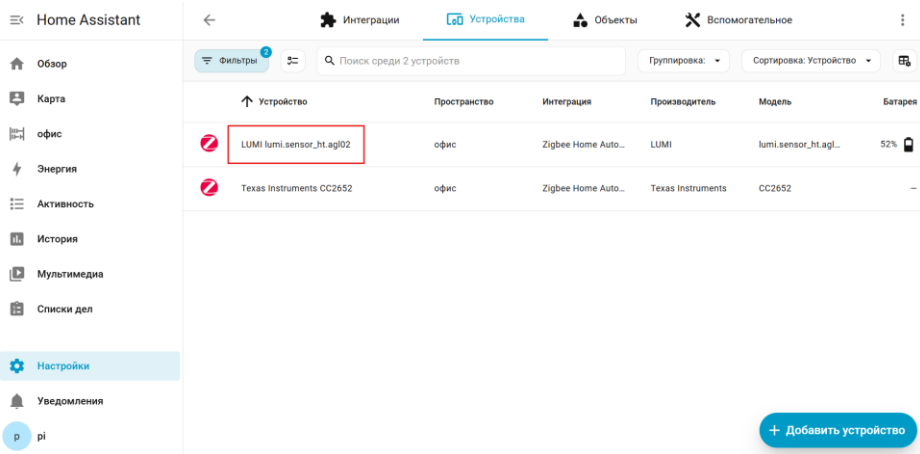


Рис. 3.26. Окно «Устройства»

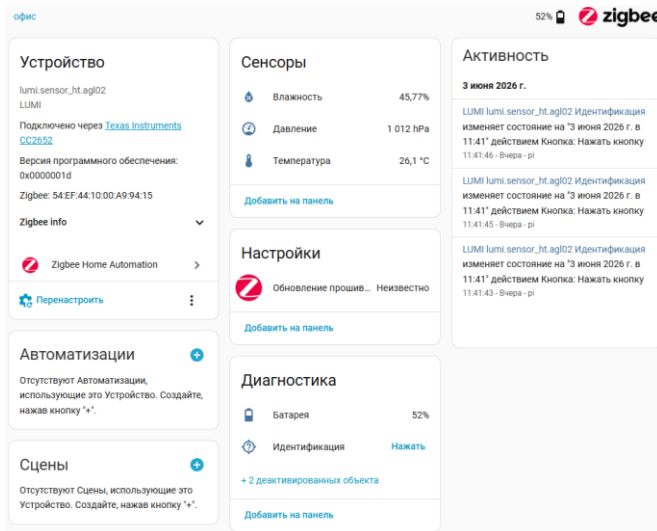


Рис. 3.27. Окно устройства

3.2.3. Zigbee2MQTT настройка

Сначала нужно включить сервис Zigbee2MQTT или убедиться, что он включен. Управлять состоянием Zigbee2MQTT можно из веб-конфигуратора шлюза.

1. Открываем веб-конфигуратор по адресу <http://IP-шлюза:8000/>. Авторизуемся. С завода авторизоваться в конфигураторе можно по логину “pi” и паролю “12345678”.
2. Переходим в раздел “Сеть” и во вкладку “Zigbee”, нажимаем на кнопку “Логирование и управление службами” (рис. 3.28).

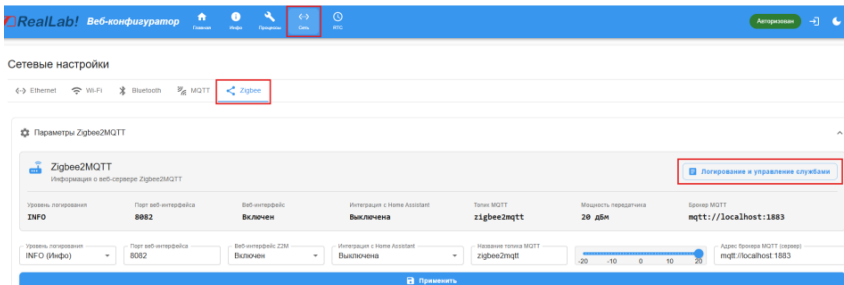


Рис. 3.28. Сравнение способов использования Zigbee

3. Подключение устройств. Работа с протоколами

3. В открытом окне нажимаем кнопку “Обновить”.

Если в окне вывода видим “-- No entries --”, сервис Zigbee2MQTT отключен.

Чтобы его включить, необходимо нажать кнопки “запустить” и “включить автозапуск” (рис. 3.29). Перезагружаем шлюз через кнопку настроек в правом верхнем углу конфигуратора - Zigbee2MQTT включён.

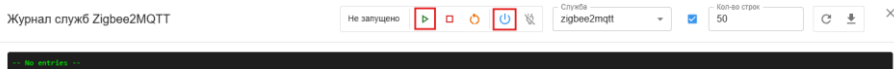


Рис. 3.29. Окно управления службой Zigbee2MQTT

4. Подключение устройств к Zigbee2MQTT производится через его собственный веб-интерфейс. Открыть его можно через веб-интерфейс шлюза в разделе “Главная” (рис. 3.30).

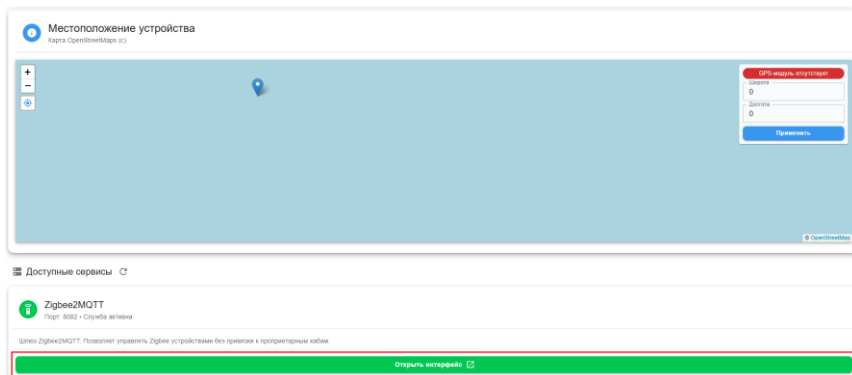


Рис. 3.30. Раздел «Главная» веб-конфигуратора

5. В веб-интерфейсе Zigbee2MQTT включаем интеграцию с Home Assistant. Для этого переходим в раздел “Настройки”, вкладку “Настройки”, подвкладку “Интеграция Home Assistant”. Ставим галочку на “enabled” (рис. 3.31).

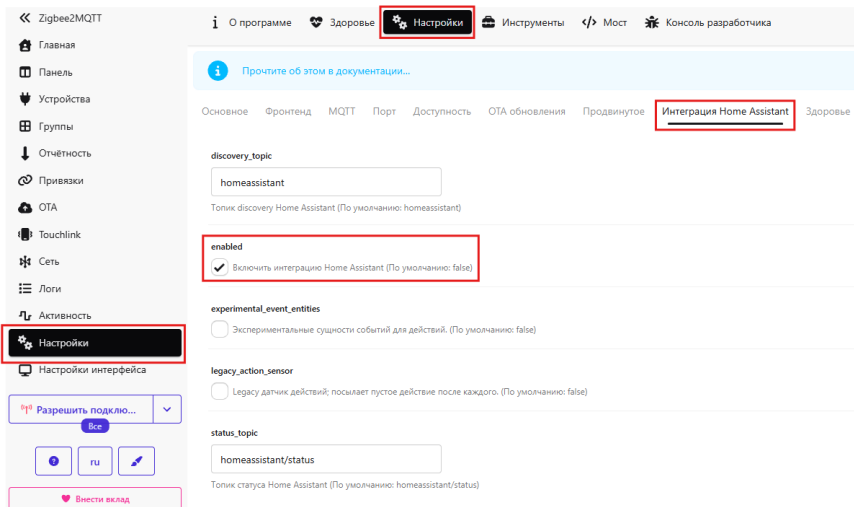


Рис. 3.31. Раздел «Настройки» Zigbee2MQTT

6. Настройка Zigbee2MQTT завершена.

3.2.4. Zigbee2MQTT подключение устройств

1. Переходим в раздел “Главная” и нажимаем “Разрешить подключение” (рис. 3.32).

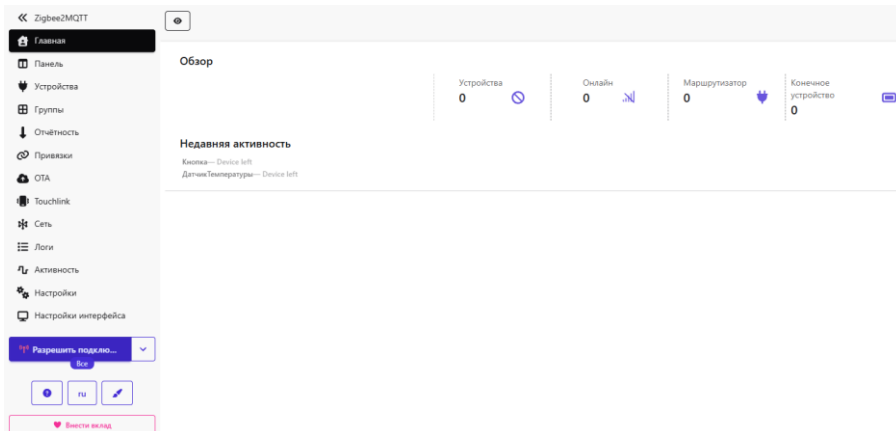


Рис. 3.32. Раздел «Главная» Zigbee2MQTT

3. Подключение устройств. Работа с протоколами

- Пока происходит поиск Zigbee устройств, переводим подключаемое устройство в состояние подключения. Например, зажимаем кнопку, пока светодиод на устройстве не начнет мигать. Можно подключать одновременно несколько устройств.

Zigbee2MQTT обнаружит устройство (рис. 3.33). Затем сервис будет ждать первой отправки пакета данных для настройки. Можно подождать или нажать кнопку принудительной отправки на Zigbee устройстве.

После этого настройка будет завершена. Данные с устройств передаются в MQTT брокер шлюза и обратно. Можно нажать “Запретить присоединение” или ничего не делать, через время присоединение запретится само.

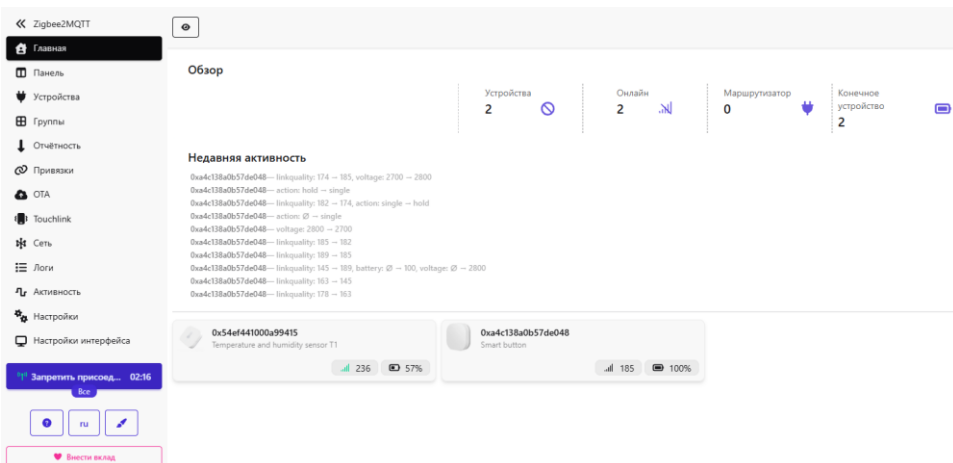


Рис. 3.33. Присоединенные устройства к Zigbee2MQTT

- Присоединенные устройства можно переименовать в разделе “Устройства” (рис. 3.34).

Подключение устройств. Работа с протоколами

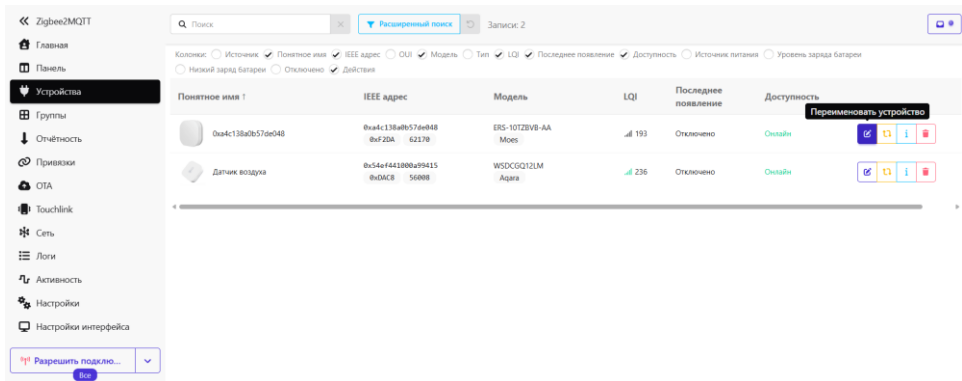


Рис. 3.34. Раздел «Устройства» Zigbee2MQTT

4. Переходим в Home Assistant (Через конфигуратор шлюза или введя `Ip_шлюза:8123` в адресную строку браузера). Нам необходимо подключиться в MQTT брокеру шлюза. Для этого перейдём в настройки -> устройства и службы (рис. 3.35).

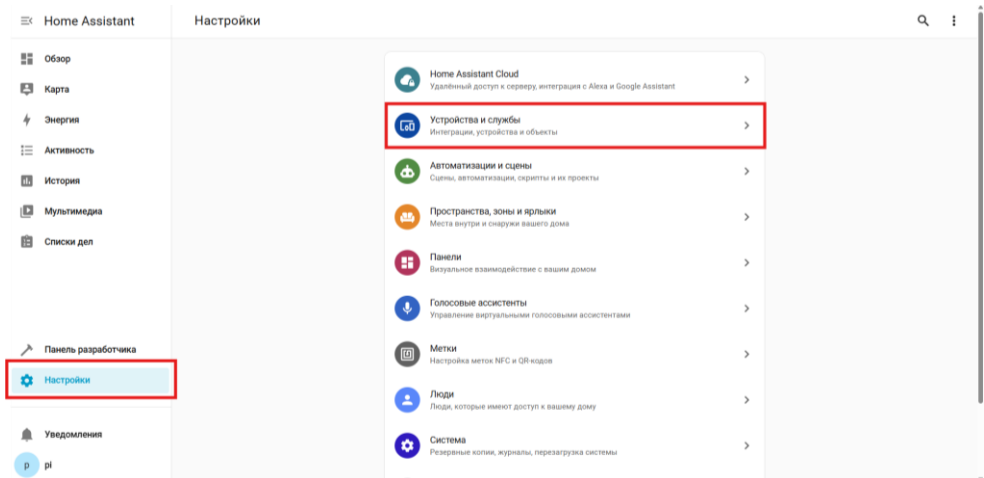


Рис. 3.35. Окно настроек Home Assistant

5. Откроется окно устройств и служб, нажмём на кнопку “Добавить интеграцию” (рис. 3.36).

3. Подключение устройств. Работа с протоколами

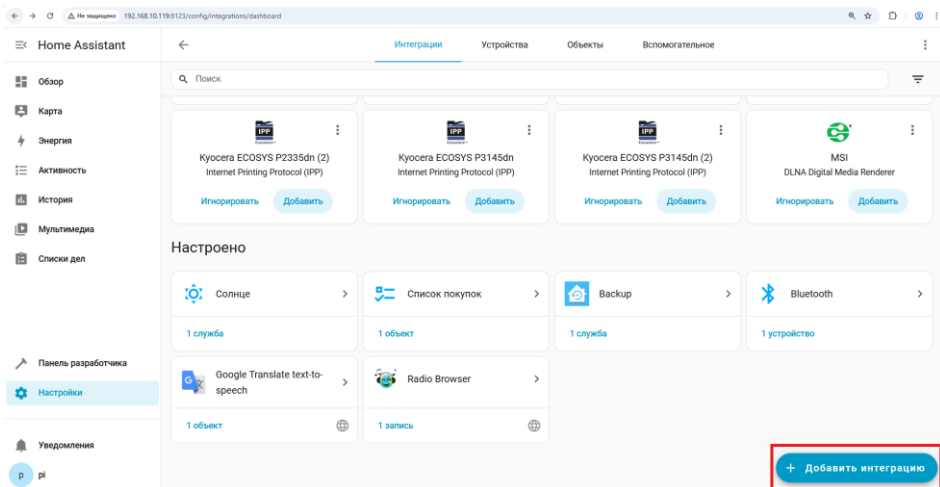


Рис. 3.36. Кнопка “Добавить интеграцию”

6. Выбираем MQTT, воспользовавшись поиском (Рис. 3.37). Затем выбираем снова “MQTT” (рис. 3.38).

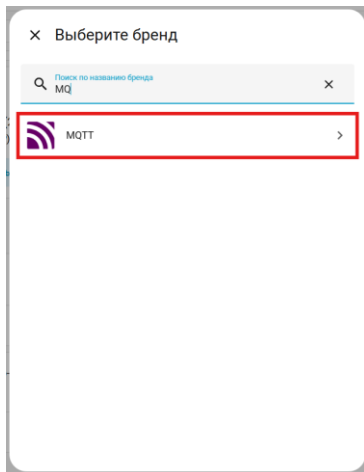


Рис. 3.37. Окно выбора устройства

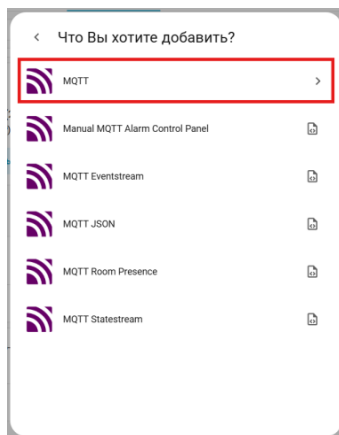
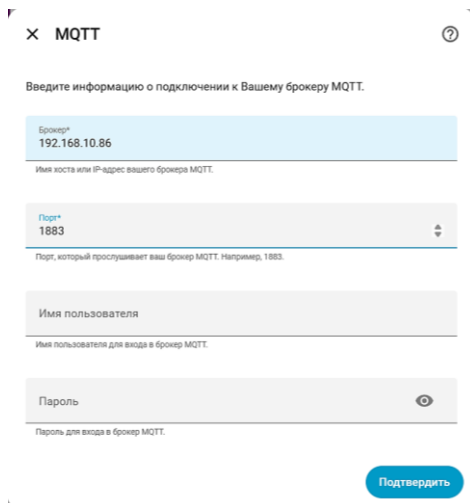


Рис. 3.38. Окно выбора устройства

7. Вводим настройки брокера (рис. 3.39):

- IP-адрес шлюза, порт = 1883
- Имя пользователя по умолчанию пустое.
- Пароль по умолчанию пустой.
- Логин и пароль брокера рекомендуется сменить.



MQTT

Введите информацию о подключении к Вашему брокеру MQTT.

Брокер*
192.168.10.86
Имя хоста или IP-адрес вашего брокера MQTT.

Порт*
1883
Порт, который прослушивает ваш брокер MQTT. Например, 1883.

Имя пользователя
Имя пользователя для входа в брокер MQTT.

Пароль
Пароль для входа в брокер MQTT.

Подтвердить

Рис. 3.39. Настройки подключения к MQTT брокеру

3. Подключение устройств. Работа с протоколами

8. Откроется меню настройки устройств, присоединенных к Zigbee2MQTT (рис. 3.40). Можно переименовать их или задать пространство. Или ничего не менять, нажав “Пропустить и завершить”.

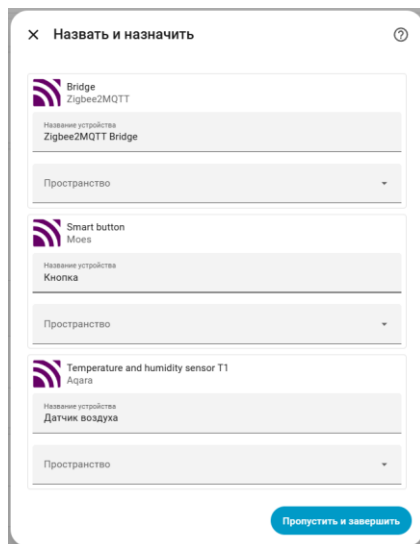


Рис. 3.40. Окно добавления найденных устройств

9. Устройства добавлены и готовы для использования в скриптах и дашбордах (рис. 3.41 - рис. 3.42).

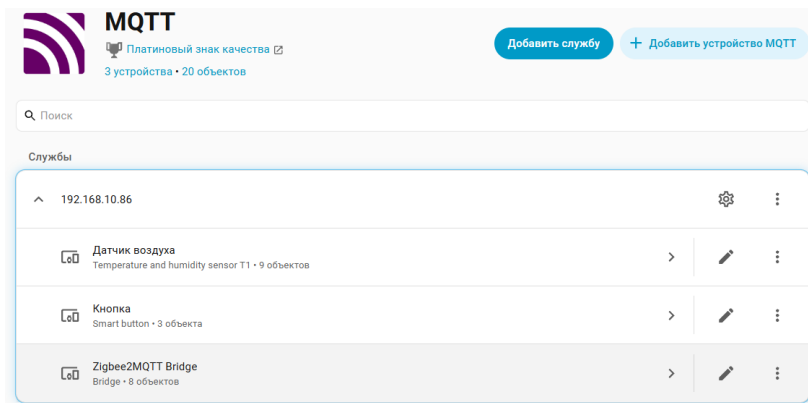


Рис. 3.41. Страница интеграции MQTT

←

Интеграции

Устройства

Объекты

Вспомогательное

Фильтры

Поиск среди 31 объекта

Группировка: Уст

Объект	Устройство	Пространство	Интеграция
Датчик воздуха			
Атмосферное давление	Датчик воздуха		MQTT
Батарея	Датчик воздуха		MQTT
Влажность	Датчик воздуха		MQTT
Датчик воздуха	Датчик воздуха		MQTT
Напряжение	Датчик воздуха		MQTT
Температура	Датчик воздуха		MQTT
Температура	Датчик воздуха		MQTT
Кнопка			
Батарея	Кнопка		MQTT
Operation mode	Кнопка		MQTT

Рис. 3.42. Окно «Объекты»

3.3. LoRaWAN

LoRaWAN (Long Range Wide Area Network) — это открытый беспроводной протокол связи, разработанный специально для создания энергоэффективных сетей большой дальности (LPWAN). Он спроектирован для интернета вещей (IoT) и межмашинного взаимодействия (M2M), где требуется передавать небольшие объемы данных на значительные расстояния с минимальным расходом энергии.

В отличие от Zigbee или Wi-Fi, LoRaWAN использует топологию «звезда из звезд» (star-of-stars). В ней нет промежуточных узлов-репитеров (маршрутизаторов), что существенно упрощает развертывание и снижает энергопотребление датчиков.

Схема движения данных выглядит следующим образом:

- Конечное устройство периодически (с заданным интервалом) опрашивает сенсоры и передают по радиоканалу LoRa радиопакеты на шлюз,

3. Подключение устройств. Работа с протоколами

содержащие идентификатор устройства, например, значения температуры, влажности, уровень заряда батареи и служебную информацию.

- Шлюз NLS-Gateway через встроенный концентратор (Packet Forwarder) принимает пакеты и передаёт их в сетевой сервер ChirpStack, где выполняется дешифрование, проверка целостности и преобразование в формат MQTT.
- Сетевой сервер ChirpStack публикует данные в MQTT-брокер в топики.
- Home Assistant выступает в роли потребителя данных, подписываясь на топики брокера MQTT, парсит JSON-сообщения и сохраняет их для последующего использования в автоматизациях или визуализации на панель.

Табл. 3. Преимущества и ограничения технологии LoRa / LoRaWAN

Параметр	Описание
Дальность связи	До 2–5 км в плотной городской застройке; до 15–20 км в условиях прямой видимости.
Автономность	Устройства класса А могут работать от одного комплекта батарей до 5–10 лет за счет глубокого спящего режима.
Проникающая способность	Частота 868 МГц (RU864/EU868) и модуляция LoRa обеспечивают уверенный прием в подвалах, шахтах и за толстыми перекрытиями.
Ограничение скорости	Низкая пропускная способность (от сотен бит/с до десятков кбит/с). Не подходит для потоковых данных или файлов.
Ограничение эфира (Duty Cycle)	Законодательные лимиты на время передачи запрещают непрерывный опрос.
Задержки (Latency)	Для устройств класса А отправка команд управления возможна только в короткие окна приема сразу после очередной передачи телеметрии.

Платформа Home Assistant не имеет встроенного радиостека LoRaWAN, поэтому интеграция выполняется по цепочке:

Датчик → LoRaWAN → Шлюз RealLab (ChirpStack) → MQTT-брокер → Home Assistant.

(Ниже процесс описан на примере LoRaWAN-датчика температуры «Вега ТД-11»).

3.3.1. Подключения LoRa датчика к ChirpStack

1. Откройте веб-конфигуратор шлюза по адресу: `http://IP_шлюза:8000/` (логин по умолчанию: `pi`, пароль: `12345678`).
2. В разделе «Главная» откройте веб-интерфейс службы ChirpStack (рис. 3.43).
3. В веб-интерфейсе сетевого сервера ChirpStack создать профиль устройства (Device Profile) и зарегистрировать приложение (Application).
4. Добавить конечное устройство, указав идентификаторы и ключи аутентификации (DevEUI, AppKey / AppSKey, NwkSKey в зависимости от режима активации OTAA или ABP).
5. В свойствах профиля устройства добавить JavaScript-код кодека (парсера) для автоматического декодирования (Payload Codec).
6. После успешного подключения датчик появится в списке активных устройств ChirpStack, а декодированные данные начнут публиковаться в локальный MQTT-брокер.

Подробное описание добавления устройств и настройки кодеков приведено в руководстве по работе с ПО шлюзов RealLab! серии NLS-Gateway на официальном сайте компании.

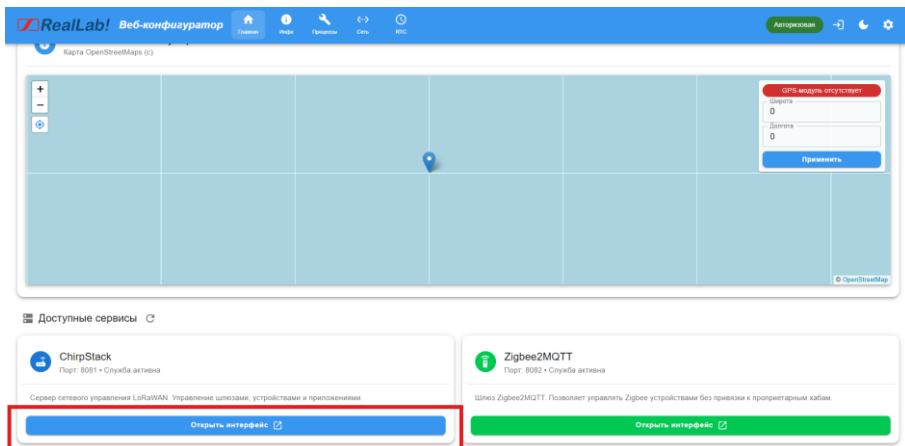


Рис. 3.43. Раздел «Главная» веб-конфигуратора

3.3.2. Получение и обработка данных LoRaWAN в Home Assistant через MQTT

1. Убедитесь, что в Home Assistant настроена интеграция MQTT (см. подраздел 3.1.1).
2. Определение топика телеметрии:
 - Откройте программу MQTT Explorer (рис. 3.44) на ПК и подключитесь к MQTT-брокеру шлюза (IP_шлюза:1883).
 - Найдите ветку событий устройства:
application/<ID_приложения>/device/<DevEUI_датчика>/event/up
 - Внутри топика сообщения передаются в формате JSON (рис. 3.45), содержащем вложенный объект object:
 - Скопируйте полный путь топика.

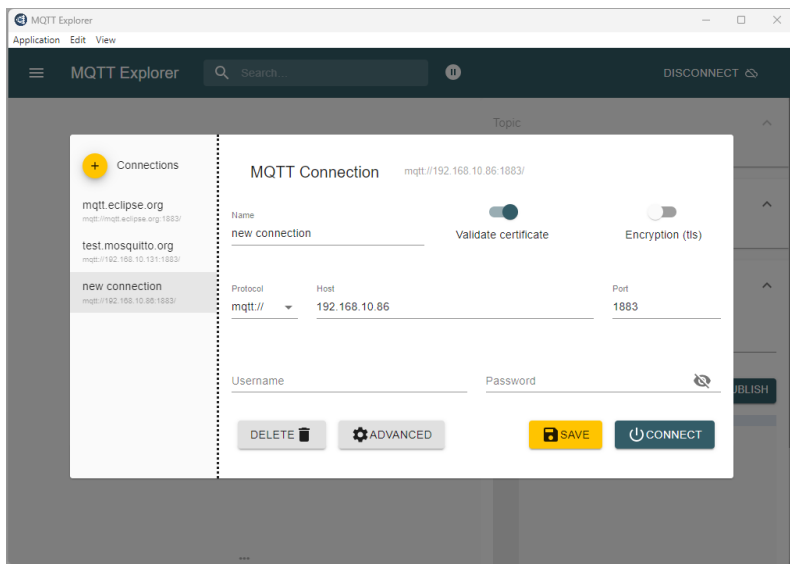


Рис. 3.44. Окно подключения MQTT Explorer

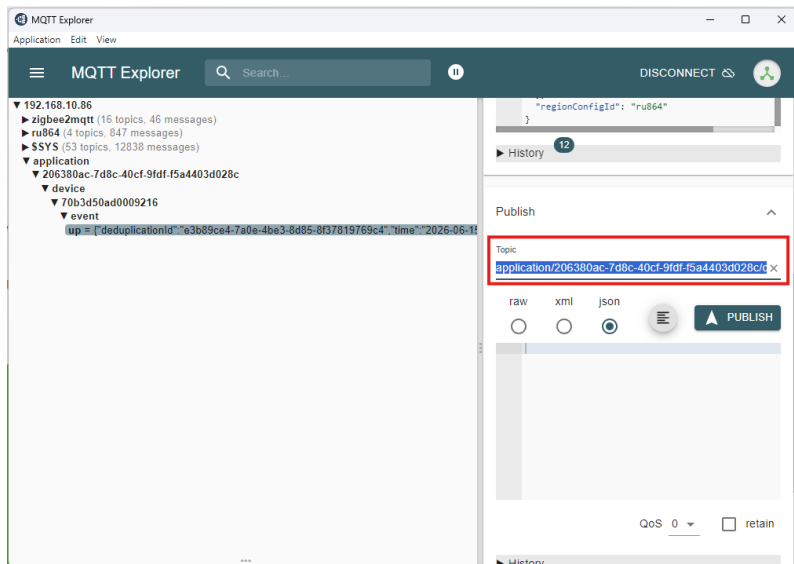


Рис. 3.45. Топик Loga-датчика в MQTT Explorer

Добавление в Home Assistant:

1. В Home Assistant перейдите в «Настройки» → «Устройства и службы» → «MQTT».
2. Нажмите «Добавить устройство MQTT вручную».
3. Задайте имя устройства (например, «Датчик Вега ТД-11») и нажмите «Далее» (рис. 3.46).

Настройте данные устройства MQTT

Введите данные устройства MQTT:

Название*
Вега ТД-11

Название вручную добавленного устройства MQTT.

Модель

Например, 'Cleanmaster Pro'.

ID модели

Например, '123NK2PRO'.

Manufacturer

E.g. Cleanmaster Ltd.

URL-адрес конфигурации

Ссылка на веб-страницу для управления конфигурацией этого устройства. Может быть URL-адресом, начинающимся с 'http://', 'https://', или внутренним URL-адресом 'homeassistant://'.

Расширенные настройки устройства

Рис. 3.46. Окно ввода имени устройства MQTT

4. Выберите тип объекта: Сенсор (Sensor) (рис. 3.47).

Настройте устройство MQTT "Вега ТД-11"

Настройте основные параметры объекта

Тип объекта*
Сенсор

Тип объекта для настройки.

Название объекта
Температура с термосопротивления

Имя объекта. Оставьте пустым, чтобы установить значение None: для отметки его как основной функции устройства MQTT.

Изображение объекта

URL-адрес изображения, которое будет назначено.

Далее

Рис. 3.47. Окно ввода типа объекта MQTT

5. Задайте параметры:

- Класс устройства (Device Class): temperature
- Единица измерения: °C
- Класс состояния (State Class): measurement

Настройте устройство MQTT "Бера ТД-11"

Пожалуйста, настройте специфические детали для объекта sensor "Бера ТД-11" Температура с термосопротивления:

Категория объекта
Диагностика

Позволяет маркировать объект устройства как конфигурационный или диагностический. Объекты из этой категории не будут передаваться компонентам облака, Alexa или Google Assistant, а также к ним не будут применяться косвенные вызовы действий, адресованные к родительским устройствам или пространствам. Сенсорам нельзя назначить класс конфигурации устройства. [Помощь](#).

Класс устройства
Температура

Класс устройства объекта sensor. [Узнать больше](#).

Класс состояния
Измерение

[Класс состояния](#) сенсора. [Узнать больше](#).

Единица измерения "°C" не поддерживается выбранным классом устройства. Пожалуйста, удалите класс устройства, выберите класс устройства, который поддерживает "°C", или выберите поддерживаемую единицу измерения из списка.

Единица измерения
°C

Определяет единицу измерения, если таковая имеется.

Дополнительные настройки

Рис. 3.48. Окно ввода категории объекта MQTT

6. В поле «Топик состояния» (State Topic) вставьте скопированный топик из ChirpStack (рис. 3.49).

3. Подключение устройств. Работа с протоколами

Настройка устройства MQTT "Bera ТД-11".

Пожалуйста, настройте специфические детали MQTT для объекта sensor "Bera ТД-11 Температура с термосопротивления".

Топик состояния*
application/206380ac-7d8c-40cf-9fdf-f5a4403d028c/device/70b3d50ad000921f

Топик MQTT, на который подписаны для получения значений состояния sensor: [Узнать больше](#).

Шаблон значения

1 {{ ... }}

Определяет [шаблон](#) для извлечения значения объекта sensor: [Подробнее](#).

Дополнительные настройки

Далее

Рис. 3.49. Настройка топика устройства

- В поле «Шаблон значения» (Value Template) введите выражение Jinja2 для извлечения нужного параметра (рис. 3.50).

Настройка устройства MQTT "Bera ТД-11".

Пожалуйста, настройте специфические детали MQTT для объекта sensor "Bera ТД-11 Температура с термосопротивления".

Топик состояния*
application/206380ac-7d8c-40cf-9fdf-f5a4403d028c/device/70b3d50ad000921f

Топик MQTT, на который подписаны для получения значений состояния sensor: [Узнать больше](#).

Шаблон значения

1 {{ value_json.object.temperature }}

Определяет [шаблон](#) для извлечения значения объекта sensor: [Подробнее](#).

Дополнительные настройки

Далее

Рис. 3.50. Настройка шаблона значения

- Один параметр датчика добавлен (Рис. 3.53). Остальные параметры добавляются аналогично.

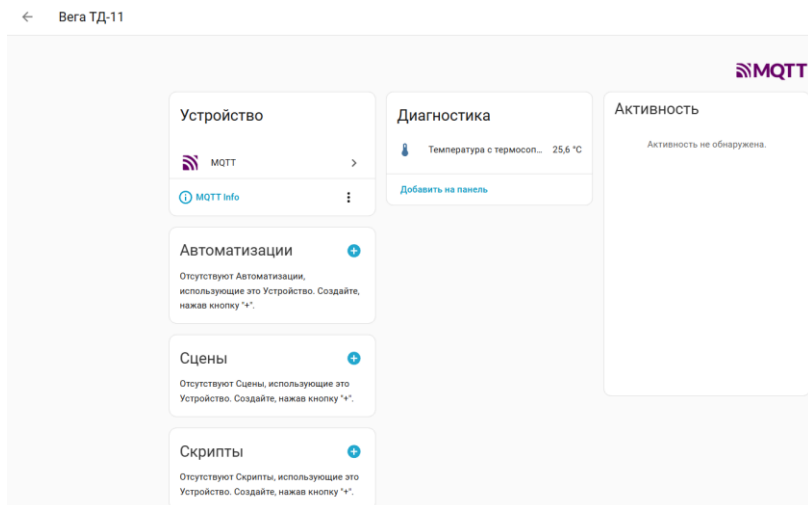


Рис. 3.51. Окно добавленного LoRa-устройства

9. Нажмите «Сохранить / Готово».

Канал температуры создан и готов для вывода на дашборд (Lovelace) или использования в автоматизациях. Аналогичным образом настраиваются сенсоры для других параметров (уровень заряда батареи, влажность и т.д.).

3.4. Modbus

Modbus – промышленный протокол связи, работающий по принципу «запрос-ответ». В зависимости от физического уровня передачи данных он разделяется на две основные версии:

- **Modbus RTU** – данные передаются в двоичном формате по последовательным линиям связи (обычно RS-485).
- **Modbus TCP** — данные упаковываются в TCP-пакеты и передаются по стандартным сетям Ethernet (обычно через порт 502).

Роли в Modbus RTU:

- **Master (Ведущий)** — инициирует обмен данными и отправляет запросы в сеть.
- **Slave (Ведомый)** — ожидает запросов к своему адресу и отправляет на них ответы.

Роли в Modbus TCP:

- Client (Клиент) — устанавливает TCP-соединение и запрашивает данные (аналог Master).
- Server (Сервер) — ожидает входящих подключений и отвечает на запросы (аналог Slave).

Примечание: В Home Assistant настройка интеграции Modbus выполняется только через конфигурационный файл `configuration.yaml`. Через веб-интерфейс доступен только просмотр и использование уже зарегистрированных объекты.

Работа Modbus описана на примере модуля дискретного вывода RealLab NLS-16DO (для Modbus RTU) и NLS-16DO-Ethernet (для Modbus TCP). Пример конфигурации доступен для скачивания на странице шлюза на официальном сайте компании RealLab.

3.4.1. Подключение к шлюзу по SSH

Для редактирования конфигурационных файлов Home Assistant рекомендуется использовать среду Visual Studio Code:

1. Запустите VS Code и установите расширение Remote - SSH (рис. 3.52).

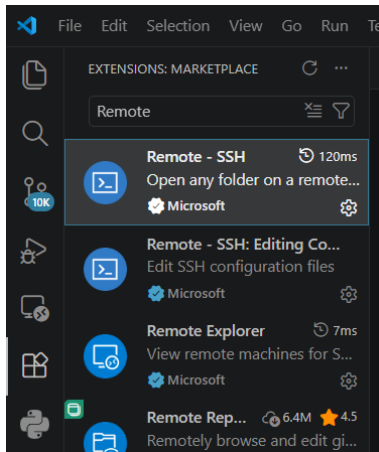


Рис. 3.52. Расширения «Remote - SSH»

2. Нажмите сочетание клавиш `Ctrl + Shift + P`, введите `Remote-SSH: Add New SSH Host...` и нажмите `Enter`.

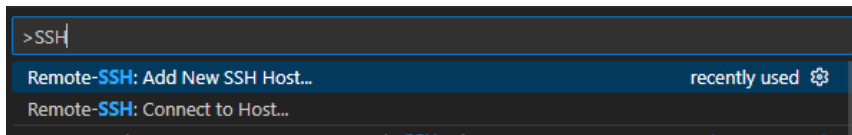


Рис. 3.53. Добавление нового SSH хоста

3. Введите строку подключения:

```
ssh pi@<IP-адрес_шлюза> -A
```

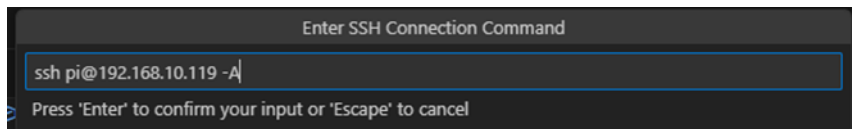


Рис. 3.54. Ввод параметров SSH хоста

4. Снова нажмите Ctrl + Shift + P, выберите Remote-SSH: Connect to Host... и выберите добавленный IP-адрес.

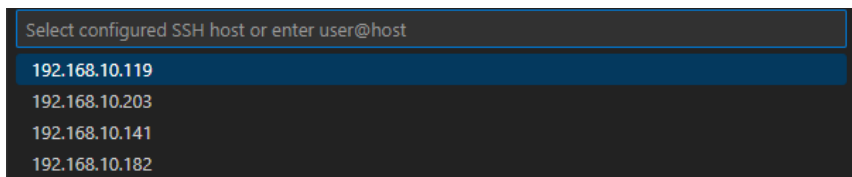


Рис. 3.55. Подключение к SSH хосту

5. При запросе типа платформы выберите Linux, логин: pi и введите пароль (по умолчанию на шлюзах Reallab: 123).

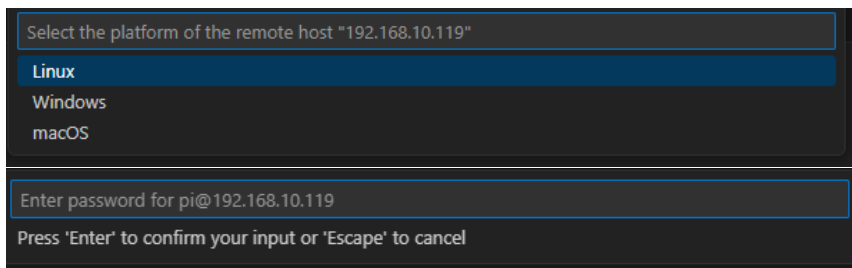


Рис. 3.56. Выбор ОС и ввод пароля

6. В проводнике VS Code нажмите «Open Folder» и укажите путь к рабочей директории Home Assistant:

```
/home/pi/homeassistant/config/
```

3. Подключение устройств. Работа с протоколами

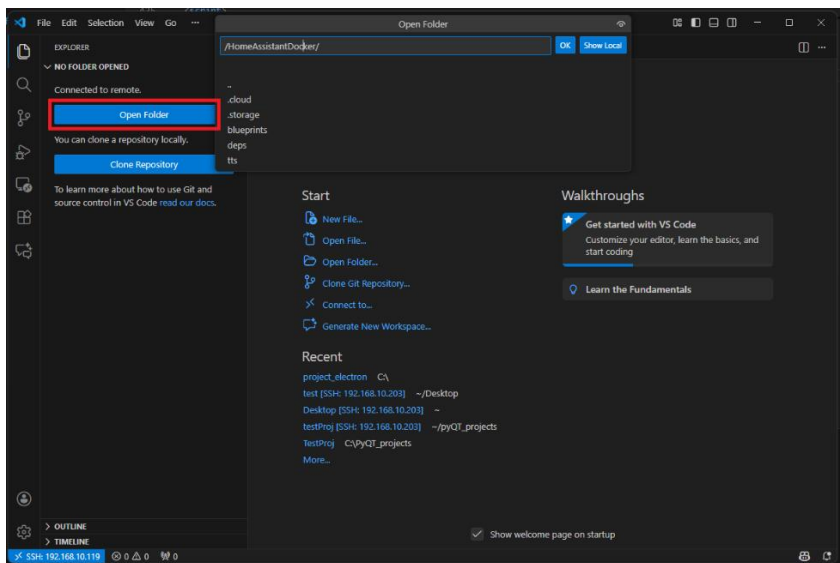


Рис. 3.57. Ввод пути до папки с конфигурационным файлом Home Assistant

7. Подтвердите доверие к файлам («Yes, I trust the authors») и откройте файл configuration.yaml (рис. 3.58).

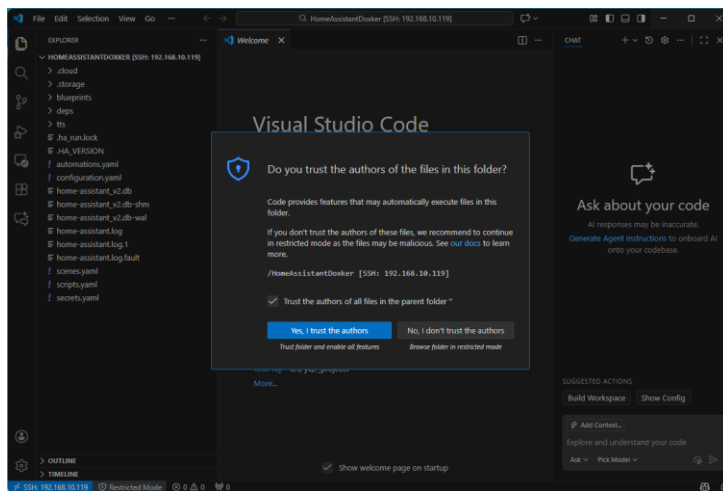


Рис. 3.58. Согласие о доверии файлам в папке

8. Открываем файл «configuration.yaml» (рис. 3.59).

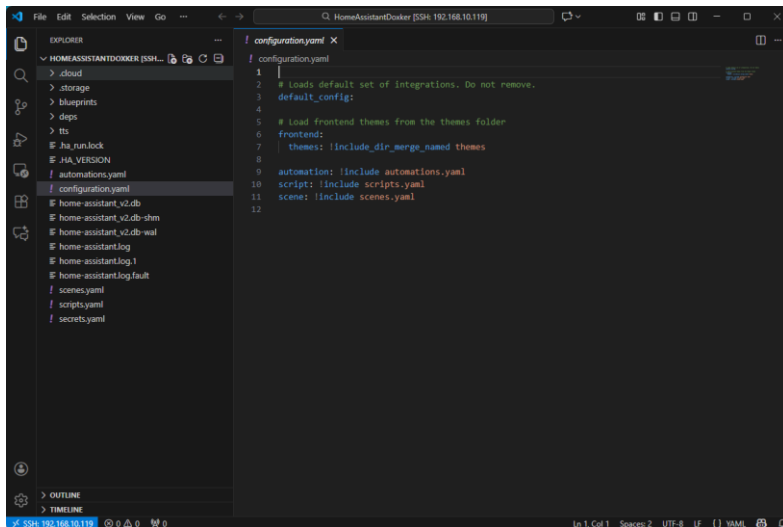


Рис. 3.59. Файл configuration.yaml

3.4.2. Настройка Modbus RTU

Подробная информация описана на сайте Home Assistant: <https://www.home-assistant.io/integrations/modbus/>.

Ниже приведен пример конфигурации для работы с модулем дискретного вывода RealLab NLS-16DO через порт RS-485 (опрос состояния и управление выходами типа Coil).

Красным текстом обозначена настройка порта. Синим текстом – настройка чтения и записи регистров.

Несколько модулей добавляются просто путем добавления новых регистров с другим id (поле slave).

Вставьте следующий блок в файл configuration.yaml:

```

# Modbus RTU
modbus:
  - name: ModbusRTUport_0
    type: serial
    port: /dev/ttyAMA2
    baudrate: 115200

```

bytesize: 8
method: rtu
parity: N
stopbits: 1
delay: 50
timeout: 5
switches:
- *name: do0_write*
 address: 15
 write_type: coil
 slave: 3
 device_class: switch
 unique_id: do0_write
- *name: do1_write*
 address: 1
 write_type: coil
 slave: 3
 device_class: switch
 unique_id: do1_write
- *name: do2_write*
 address: 2
 write_type: coil
 slave: 3
 device_class: switch
 unique_id: do2_write
- *name: do3_write*
 address: 3
 write_type: coil
 slave: 3
 device_class: switch
 unique_id: do3_write
binary_sensors:
- *name: do0_read*
 address: 0
 input_type: coil
 scan_interval: 10
 slave: 3
 unique_id: do0_read

3.4.3. Настройка Modbus TCP

Ниже приведен пример опроса модуля дискретного вывода RealLab NLS-16DO-Ethernet по сети Ethernet.

Вставьте следующий блок в файл configuration.yaml:

Modbus TCP

modbus:

- name: NLS-16DO-Ethernet

type: tcp

host: 192.168.10.221 # IP-адрес самого I/O-модуля (не ПЛК!)

port: 502

switches:

- name: do0_write

address: 0

write_type: coil

slave: 1

device_class: switch

unique_id: do0_write

- name: do1_write

address: 1

write_type: coil

slave: 1

device_class: switch

unique_id: do1_write

- name: do2_write

address: 2

write_type: coil

slave: 1

device_class: switch

unique_id: do2_write

- name: do3_write

address: 3

write_type: coil

slave: 1

device_class: switch

unique_id: do3_write

binary_sensors:

- name: do0_read

address: 0

input_type: coil

scan_interval: 100

slave: 1
unique_id: do0_read

3.4.4. Применение конфигурации

1. Сохраните изменения в файле configuration.yaml (Ctrl + S).
2. В веб-интерфейсе Home Assistant перейдите в «Инструменты разработчика» → «YAML».
3. Нажмите кнопку «Проверить конфигурацию». Убедитесь, что синтаксических ошибок нет («Конфигурация верна»).
4. Перезапустите Home Assistant («Настройки» → «Система» → кнопка перезапуска в правом верхнем углу).
5. После перезапуска добавленные сущности появятся в разделе «Настройки» → «Устройства и службы» → «объекты» и станут доступны для добавления на дашборд.

4. Программирование алгоритмов и автоматизация

Данный раздел посвящен созданию логики управления и автоматизации в среде Home Assistant. Система предоставляет несколько инструментов для реализации алгоритмов любой сложности: от простых правил «условие-действие» до сложных промышленных сценариев с использованием визуального программирования в Node-RED.

4.1. Инструменты автоматизации в Home Assistant

В Home Assistant реализована многоуровневая система создания логики, включающая четыре основных инструмента. Выбор конкретного инструмента зависит от сложности задачи, необходимости повторного использования и требуемого уровня гибкости.

4.1.1. Автоматизации (Automations)

Автоматизация — это основной инструмент для создания правил, работающих по принципу «триггер — условие — действие». Автоматизации выполняются в фоновом режиме и реагируют на события без участия пользователя.

Структура автоматизации:

- **Триггер** — событие, которое инициирует запуск автоматизации (например, достижение заданного времени, изменение состояния датчика, получение веб-хука).
- **Условие** — необязательная проверка, которая должна быть выполнена для продолжения работы автоматизации после срабатывания триггера (например, «только если на улице темно»).
- **Действие** — команда, которая выполняется при срабатывании триггера и выполнении всех условий (например, включить реле, отправить уведомление).

Назначение: автоматизации предназначены для создания автономных правил, которые работают непрерывно и не требуют ручного запуска.

Где создавать: Настройки → Автоматизации и сцены → Автоматизации → Создать автоматизацию.

4.1.2. Скрипты (Scripts)

Скрипт представляет собой заранее определенную последовательность действий, которая выполняется шаг за шагом. В отличие от автоматизации, скрипт не имеет триггера и не может запускаться автоматически по событию.

Ключевые особенности:

- Скрипты запускаются вручную из интерфейса (по нажатию кнопки) или вызываются из других автоматизаций и скриптов.
- Скрипты поддерживают паузы (delay), условные операторы (choose), повторение действий (repeat) и параллельное выполнение.
- Скрипты позволяют избежать дублирования кода: если одна и та же последовательность действий используется в нескольких автоматизациях, ее целесообразно вынести в отдельный скрипт.

Назначение: создание многократно используемых последовательностей действий и ручное управление группами устройств.

Где создавать: Настройки → Автоматизации и сцены → Скрипты → Создать скрипт.

4.1.3. Сцены (Scenes)

Сцена — это «снимок» состояния группы устройств в определенный момент времени. При активации сцены все входящие в нее устройства мгновенно переходят в сохраненные состояния.

Пример использования:

- Сцена «Вечерний киносеанс» — яркость ламп 20%, цвет торшера — теплый белый, шторы закрыты.
- Сцена «Уход из дома» — все светильники выключены, термостат переведен в экономичный режим, все розетки отключены.

Назначение: быстрое переключение группы устройств в заданное состояние одним действием.

Где создавать: Настройки → Автоматизации и сцены → Сцены → Создать сцену.

4.1.4. Чертежи (Blueprints)

Чертеж (Blueprint) — это готовый шаблон автоматизации, созданный сообществом Home Assistant. Чертеж содержит всю логику правила, но оставляет возможность пользователю выбрать конкретные устройства (сущности), которые будут использоваться.

Преимущества:

- Ускорение разработки типовых сценариев (например, управление светом по датчику движения).
- Снижение вероятности ошибок — готовая логика уже проверена другими пользователями.
- Возможность создавать собственные чертежи для повторного использования внутри организации.

Где создавать: Настройки → Автоматизации и сцены → Чертежи → Импортировать чертеж или создать чертеж.

4.2. Создание автоматизаций: от простого к сложному

Данный подраздел описывает процесс создания автоматизаций — от использования визуального редактора до написания сложной логики на языке YAML.

4.2.1. Визуальный редактор: триггеры, условия, действия

Home Assistant предоставляет интуитивно понятный визуальный редактор для создания автоматизаций без необходимости написания кода. Редактор

доступен в разделе «Автоматизации» и позволяет добавлять, настраивать и упорядочивать триггеры, условия и действия через графический интерфейс.

Порядок создания автоматизации в визуальном редакторе:

1. Перейдите в раздел Настройки → Автоматизации и сцены → Автоматизации.
2. Нажмите кнопку «Создать автоматизацию».
3. Выберите один из предложенных вариантов:
 - «Начать с пустой автоматизации» — создание правила с нуля.
 - «Использовать чертёж» — использование готового шаблона из библиотеки.
4. Введите название автоматизации (обязательное поле).
5. Добавьте один или несколько триггеров:
 - Нажмите «Добавить триггер».
 - Выберите тип триггера (например, «Состояние», «Время», «Событие»).
 - Настройте параметры выбранного триггера (сущность, порог срабатывания, время и т.д.).
6. Добавьте условия (опционально):
 - Нажмите «Добавить условие».
 - Выберите тип условия (например, «Состояние сущности», «Время»).
 - Настройте параметры условия.
7. Добавьте одно или несколько действий:
 - Нажмите «Добавить действие».
 - Выберите тип действия (например, «Вызвать службу», «Установить состояние»).
 - Настройте параметры действия (выберите сущность и действие).
8. Настройте дополнительные параметры (необязательно):

- **Режим выполнения** — определяет поведение автоматизации при повторных срабатываниях (см. раздел 4.2.3).
- **Длительность** — задержка перед выполнением действий.
- **Максимальное количество срабатываний** — ограничение на число выполнений.

9. Нажмите «Сохранить».

10. Включите автоматизацию с помощью переключателя на карточке созданного правила.

Примечание: после сохранения автоматизация отображается в списке. Вы можете в любой момент редактировать ее параметры, отключать, удалять или просматривать журнал выполнения (trace).

4.2.2. Основные типы триггеров

Триггер является обязательным элементом любой автоматизации. Ниже перечислены наиболее часто используемые типы триггеров с кратким описанием и примерами.

Триггер «Состояние» (State). Срабатывает при изменении состояния выбранной сущности.

Параметры: Entity ID (идентификатор сущности), атрибут (опционально), условия изменения (от, до, в течение).

Пример: «Датчик температуры изменил значение с 22.5 °C на 23.0 °C».

Триггер «Числовое состояние» (Numeric State). Срабатывает при пересечении заданного числового порога.

Параметры: Entity ID, верхний/нижний порог, условие (выше, ниже, равно).

Пример: «Температура превысила 30 °C».

Триггер «Время» (Time). Срабатывает в указанное время (абсолютное или относительное). Варианты:

- **Фиксированное время** — срабатывание в конкретное время (например, 08:00).
- **Закат/Восход** — срабатывание относительно времени восхода или заката солнца (с возможностью смещения).

- Периодический — срабатывание с заданным интервалом (например, каждые 5 минут).

Пример: «Запуск в 07:30 каждый день».

Триггер «Событие» (Event). Срабатывает при получении определенного события от системы или внешнего устройства.

Параметры: Тип события (например, click от кнопки Zigbee), дополнительные данные события.

Пример: «Двойное нажатие на кнопку Zigbee».

Триггер «Шаблон» (Template). Срабатывает при выполнении заданного условия на языке Jinja2.

Параметры: Шаблон, возвращающий булево значение (true/false).

Пример: `{{ states('sensor.temperature') | float > 25 and states('sensor.humidity') | float < 40 }}`.

Триггер «Домашний помощник» (Home Assistant). Срабатывает при системных событиях (запуск, остановка, обновление).

Пример: «Запустить автоматизацию при старте Home Assistant».

Триггер «Веб-хук» (Webhook). Срабатывает при получении HTTP-запроса на указанный URL.

Назначение: используется для интеграции с внешними системами, например, для запуска автоматизации из мобильного приложения, внешнего сервиса или по команде от Node-RED.

Пример: «Запустить скрипт по GET-запросу от внешней системы мониторинга».

4.2.3. Режимы выполнения автоматизаций

Режим выполнения определяет поведение автоматизации при многократных срабатываниях до завершения предыдущего выполнения. Выбор режима (см. табл. 4) критически важен для корректной работы систем, где возможны частые повторные срабатывания (например, датчик движения).

4. Программирование алгоритмов и автоматизация

Табл. 4. Режимы выполнения автоматизаций

Режим	Описание	Рекомендации по применению
single (одионочный)	Новое срабатывание игнорируется, если автоматизация уже выполняется.	Использовать по умолчанию для большинства сценариев.
restart (перезапуск)	Текущее выполнение прерывается, запускается новое выполнение.	Применять для сценариев, где важна актуальность последнего события (например, управление кондиционером по температуре).
queued (очередь)	Новые срабатывания помещаются в очередь и выполняются последовательно после завершения предыдущих.	Использовать, если важно обработать все события (например, архивация данных).
parallel (параллельный)	Новые выполнения запускаются параллельно с текущими (без ограничений).	Применять для независимых задач, когда одновременный запуск нескольких экземпляров допустим.

Настройка: выбор режима осуществляется в визуальном редакторе в разделе «Дополнительные параметры» (или через атрибут `mode` в YAML).

4.2.4. Редактирование автоматизаций в YAML

Для сложных сценариев, где визуальный редактор недостаточно гибок (например, при использовании вложенных условий, циклов или сложных шаблонов), доступно редактирование автоматизаций в формате YAML.

Где находятся YAML-файлы:

- Основные автоматизации хранятся в файле `automations.yaml` в папке конфигурации Home Assistant.
- Скрипты хранятся в файле `scripts.yaml`.
- Сцены хранятся в файле `scenes.yaml`.

- Папка конфигурации Home Assistant (/home/pi/homeassistant/config/).

Рекомендации по работе с YAML:

- Соблюдайте корректные отступы (2 пробела на каждый уровень вложенности).
- Используйте проверку конфигурации перед перезапуском.
- Регулярно создавайте резервные копии файлов конфигурации.

4.3. Расширенная логика с Node-RED

Node-RED представляет собой среду визуального программирования, ориентированную на создание потоков данных и интеграцию разнородных систем. В связке с Home Assistant Node-RED выступает в роли внешнего движка автоматизаций, позволяя реализовывать логику, недоступную или неудобную для реализации встроенными средствами.

Платформа Node-RED позволяет удобно работать с различными протоколами, которые не поддерживает Home Assistant: Modbus (RTU/TCP), OPC UA, CAN, IEC.

Более подробная информация о программировании шлюза ReLLab в среде Node-RED, смотрите в руководство по программированию шлюзов в среде Node-RED на странице шлюзов с поддержкой Node-RED на сайте компании RealLab!.

Компоненты в связке Home Assistant и Node-RED:

- **Home Assistant** - выступает в роли единого интерфейса управления, базы данных состояний и координатора событий.
- **Node-RED** - выполняет функции внешнего движка автоматизаций, обрабатывая сложную логику, циклы, таймеры и вычисления.
- **Интеграция Node-RED Companion в Home Assistant** - связующее звено, которое обеспечивает двустороннюю синхронизацию и позволяет создавать виртуальные сущности (например, сенсоры или переключатели) в Home Assistant напрямую из среды Node-RED.

4.3.1. Установка и настройка Node-RED и интеграции с Home Assistant

На шлюзах RealLab серии NLS-Gateway с суффиксом -HA среда Node-RED не предустановлена. Для ее использования необходимо выполнить установку.

Способ установки через дополнения (Add-ons) Home Assistant:

1. В интерфейсе Home Assistant перейдите в Настройки → Дополнения.
2. Нажмите кнопку «Магазин дополнений» (Add-on Store).
3. Найдите и установите дополнение «Node-RED» (разработчик: Home Assistant Community).
4. Запустите дополнение и перейдите на его веб-интерфейс (обычно по адресу http://IP_шлюза:1880/).

Примечание: дополнение «Node-RED» в интерфейсе Home Assistant по умолчанию установлено.

Настройка интеграции Node-RED с Home Assistant

1. В интерфейсе Node-RED откройте меню (три полосы справа) → «Палитра» → «Установить».
2. Установите пакет `node-red-contrib-home-assistant-websocket` (в поиске введите `home-assistant-websocket`).
3. После установки в левой панели Node-RED появятся ноды группы «Home Assistant».
4. Добавьте ноду `server` на рабочую область и дважды кликните по ней для настройки.
5. Укажите параметры подключения к Home Assistant:
 - Base URL: `http://IP_шлюза:8123` (URL вашего Home Assistant).
 - Access Token: создайте долгосрочный токен доступа в Home Assistant (Настройки → Пользователи → Ваш профиль → Долгосрочные токены доступа → Создать).
 - Порт WebSocket: оставьте по умолчанию (8123).
6. Нажмите «Готово». Если подключение успешно, на ноду `server` загорится зеленый индикатор.

Готово: Node-RED успешно подключен к Home Assistant и готов к созданию потоков автоматизации.

Пример с подробной инструкцией по установке и интеграции Node-RED с Home Assistant доступен для скачивания на соответствующей странице шлюза на сайте компании RealLab!.

5. Визуализация в home assistant

5.1. Общие сведения о системе визуализации

Интерфейс пользователя Home Assistant построен на базе Lovelace — гибкой и адаптивной системы визуализации, которая позволяет создавать индивидуальные панели управления (дашборды) для мониторинга и управления подключенными устройствами.

Ключевые возможности Lovelace:

Адаптивность: интерфейс оптимизирован для отображения на персональных компьютерах, планшетах и мобильных устройствах.

Кастомизация: пользователь может создавать неограниченное количество панелей с различным набором карточек, макетом и уровнем доступа.

Гибкость: поддерживаются различные типы карточек (плитки, графики, кнопки, выпадающие списки, ввод чисел и др.), которые можно комбинировать в разделы и группы.

Режимы редактирования: доступны визуальный редактор (для начинающих) и редактор YAML-кода (для опытных пользователей).

По умолчанию в системе существует одна панель — «Обзор» (Overview), на которой автоматически отображаются все обнаруженные устройства и сущности. Однако для удобства оператора рекомендуется создавать специализированные панели, сгруппированные по функциональному назначению (например, «Климат-контроль», «Энергомониторинг», «Управление освещением»).

5.2. Создание новой панели управления

Для создания пользовательской панели выполните следующие действия:

1. В боковом меню навигации перейдите в раздел «Настройки» (Settings) (рис. 5.1).

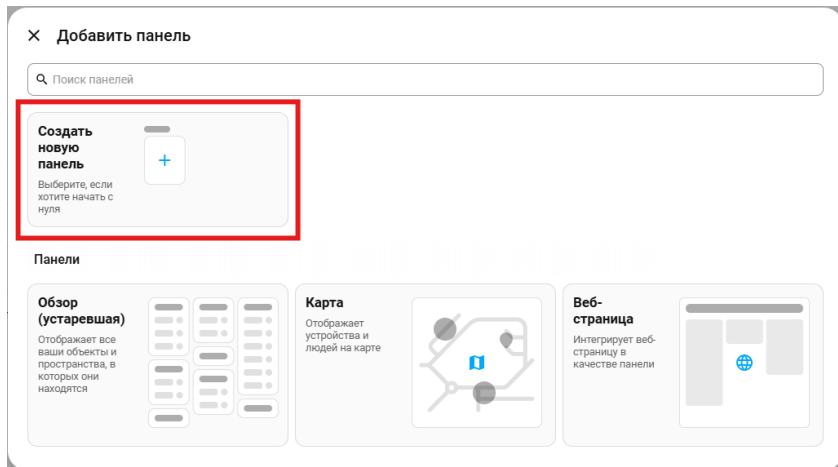


Рис. 5.1. Окно создания панели

2. На выбор представлено несколько типов панелей. Создадим новую пустую панель (рис. 5.1).
3. Придумать произвольное название панели, нажать «Создать» (рис. 5.2).

Рис. 5.2. Ввод названия и выбор иконки панели

4. Панель появится в списке панелей и в левом столбце (рис. 5.3).

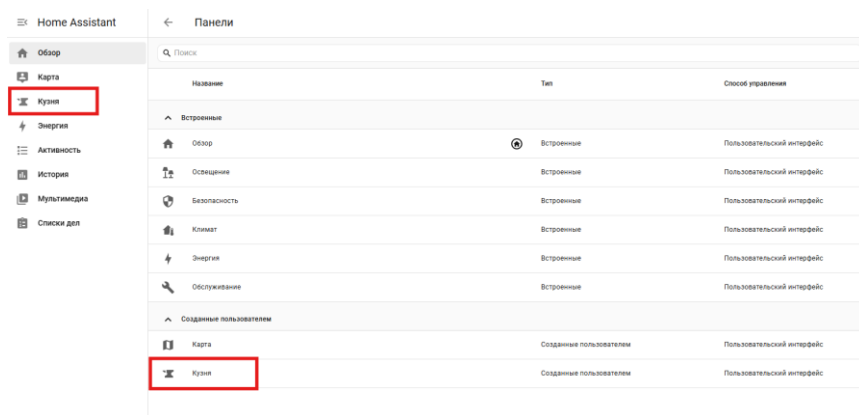


Рис. 5.3. Добавленная панель в списке панелей Home Assistant

- Откроем панель, нажав на нее в левом столбце. На панели нажмём символ карандаша сверху справа (рис. 5.4), чтобы открыть режим редактирования.

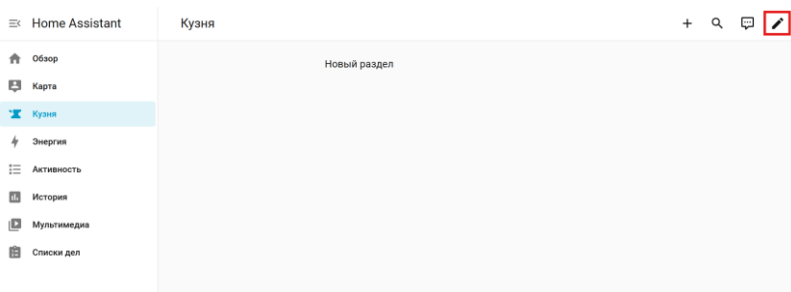


Рис. 5.4. Кнопка редактирования панели

- Открыт редактор панели управления, позволяющий гибко настроить интерфейс. Рассмотрим добавление двух объектов – температуру с датчика и регулировку силы тока.
- Нажмём кнопку «плюс» внутри раздела «Новый раздел» (рис. 5.5).

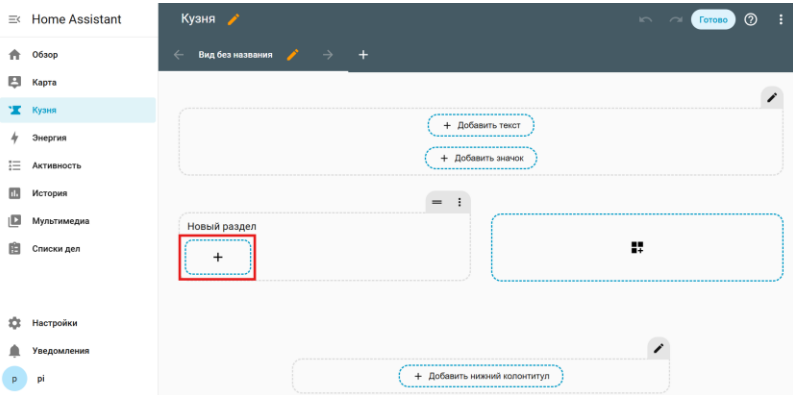


Рис. 5.5. Кнопка добавления объекта на панель

8. Выберем «Устройства» → Термометр → «Температура» → «Плитка – График тренда» (рис. 5.6).

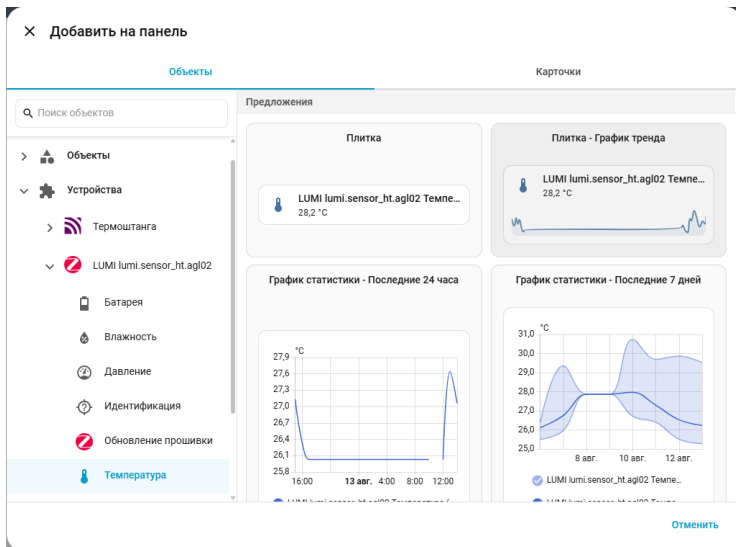


Рис. 5.6. Окно выбора добавляемого объекта и вида плитки отображения

9. Нажмём снова кнопку «плюс» внутри раздела «Новый раздел» (рис. 5.7).
10. Выберем «Устройства» → NLS-4AO → «АО0» → «Плитка – Ввод числа».

11. На меню редактора нажать кнопку «Готово» сверху справа экрана.
12. На панель управления добавлено 2 объекта. По тому же принципу можно добавить заголовки или разделить объекты на группы.

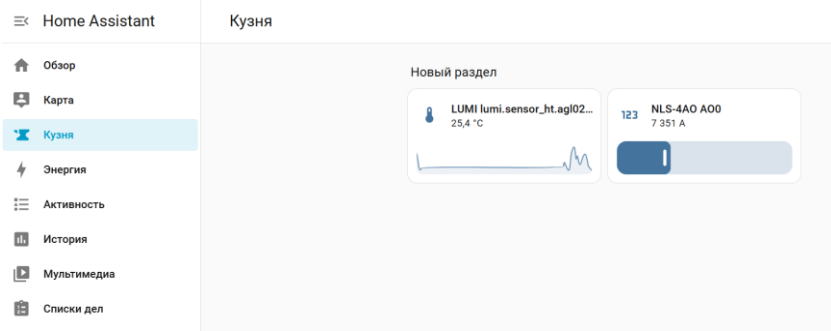


Рис. 5.7. Окно выбора добавляемого объекта и вида плитки отображения

5.3. Навигация и режимы работы с панелью

При открытии созданной панели пользователь имеет доступ к двум основным режимам работы:

- **Режим просмотра** — стандартный режим, в котором отображаются все добавленные карточки. Данный режим предназначен для мониторинга и управления устройствами конечным пользователем.
- **Режим редактирования** — режим конфигурирования панели, доступный пользователям с правами администратора. Для перехода в режим редактирования нажмите кнопку с изображением карандаша (✎) в правом верхнем углу панели.

В режиме редактирования доступны следующие действия:

- добавление, удаление и перемещение карточек;
- настройка внешнего вида и поведения каждой карточки;
- изменение структуры панели (разделы, группы, столбцы);
- импорт и экспорт конфигурации панели в формате YAML.

5.4. Добавление и настройка карточек

Карточки (Cards) являются основными элементами визуализации в Lovelace. Каждая карточка отображает одну сущность (или группу сущностей) и определяет способ ее представления — в виде плитки, графика, переключателя, ползунка и т.д.

5.4.1. Добавление новой карточки

1. Войдите в режим редактирования панели (кнопка с карандашом).
2. В рабочей области панели нажмите кнопку «Добавить карточку» (Add Card) или значок «+» внутри раздела.
3. Откроется диалоговое окно выбора типа карточки и сущности:
 - Поиск сущности — введите название сущности или ее идентификатор (Entity ID) для быстрого поиска.
 - Выбор из списка — разверните меню «Устройства» (Devices) и выберите нужное устройство, затем конкретную сущность из списка доступных параметров.
4. После выбора сущности система автоматически предложит наиболее подходящий тип карточки (например, для датчика температуры — плитку с графиком тренда). Вы можете выбрать другой тип, нажав на иконку текущего типа и выбрав вариант из выпадающего списка.
5. Настройте дополнительные параметры карточки (при необходимости):
 - Название (Name) — задайте отображаемое имя для карточки (если оно должно отличаться от имени сущности).
 - Вид (Display) — выберите стиль отображения (плитка, график, датчик, кнопка и т.д.).
 - Дополнительные параметры — в зависимости от типа сущности могут быть доступны настройки диапазона значений, единиц измерения, цвета фона, порогов срабатывания и др.
6. Нажмите кнопку «Сохранить» (Save) для добавления карточки на панель.

5.4.2. Пример добавления карточек

1. В режиме редактирования нажмите «+».

2. В поле поиска введите temperature или выберите в меню «Устройства» нужный термометр.
3. Выберите сущность (например, sensor.temperature_sensor_1).
4. В качестве типа карточки выберите «Плитка – График тренда» (Tile – Trend Graph) для отображения как текущего значения, так и динамики изменения.
5. При необходимости задайте название, например, «Температура в цехе №3».
6. Нажмите «Сохранить».

6. Работа с модулями ввода и вывода reallab

В зависимости от аппаратной модификации модули ввода-вывода компании RealLab обеспечивают поддержку проводных интерфейсов Ethernet и RS-485, а также работу по протоколам MQTT, Modbus TCP, Modbus RTU и беспроводному радиоканалу LoRaWAN.

6.1. Модули серии NLS xx Ethernet с поддержкой MQTT

Модули дискретного и аналогового ввода-вывода серии NLS-Ethernet (например, NLS-16DI-Ethernet, NLS-16DO-Ethernet, NLS-8AI-Ethernet и др.) оснащены встроенным MQTT-клиентом. Это позволяет передавать телеметрию напрямую в брокер сообщений шлюза или в облачную платформу без постоянного циклического опроса (передача выполняется по таймеру).

6.1.1. Предварительное конфигурирование модуля

Для организации работы с модулем по протоколу MQTT необходимо выполнить предварительное конфигурирование:

1. Подключитесь к встроенному веб-интерфейсу модуля NLS через браузер (по заводскому IP-адресу, например, 192.168.0.1).
2. В разделе «Сетевые настройки MQTT» задайте параметры подключения:
 - Broker IP / Host: IP-адрес шлюза NLS-Gateway (или локальный IP брокера);
 - Port: порт брокера (по умолчанию 1883);
 - Цикл отправления данных.



Состояние дискретных входов	Имя модуля: NLS-32DI-Ethernet	MAC: 70.B3.D5.22.30.00	Версия ПО: 15.04.2026																
Настройки	Настройки модуля																		
Настройки MQTT	REALLAB/NLS32DIE/70B3D5223000/STATUS																		
Изменение логина и пароля	<table border="1"> <thead> <tr> <th></th> <th>Изменения</th> </tr> </thead> <tbody> <tr> <td>MQTT</td> <td>☒</td> </tr> <tr> <td>URL Брокера:</td> <td> broker.hivemq.com </td> </tr> <tr> <td>IP-адрес Брокера:</td> <td> 192 168 0 77 </td> </tr> <tr> <td>Порт:</td> <td> 1883 </td> </tr> <tr> <td>Цикл отправки сообщений, с.:</td> <td> 5 </td> </tr> <tr> <td>Keep Alive, с.:</td> <td> 60 </td> </tr> <tr> <td>Статус: online</td> <td>☐</td> </tr> </tbody> </table>				Изменения	MQTT	☒	URL Брокера:	broker.hivemq.com	IP-адрес Брокера:	192 168 0 77	Порт:	1883	Цикл отправки сообщений, с.:	5	Keep Alive, с.:	60	Статус: online	☐
	Изменения																		
MQTT	☒																		
URL Брокера:	broker.hivemq.com																		
IP-адрес Брокера:	192 168 0 77																		
Порт:	1883																		
Цикл отправки сообщений, с.:	5																		
Keep Alive, с.:	60																		
Статус: online	☐																		
	Переконфигурировать модуль																		

Рис. 6.1. Меню веб-интерфейса «Настройки MQTT»

Примечание: более подробная информация о конфигурировании и эксплуатации модулей приведена в эксплуатационной документации на странице конкретной модели на официальном сайте компании.

6.1.2. Приём данных в Home Assistant

Для приёма данных в IoT-платформе Home Assistant добавьте интеграцию «MQTT» из списка интеграций, укажите параметры подключения к брокеру и добавьте MQTT устройство с целевым топиком, после чего будет создано устройство и объект для получаемого значения. Более подробная информация о работе с протоколом MQTT и примеры создания потоков опроса приведены в подразделе 3.1.

6.2. Модули с поддержкой Modbus RTU/TCP

Модули дискретного и аналогового ввода-вывода серии NLS (как в исполнении с интерфейсом RS-485, так и в исполнении Ethernet) поддерживают стандартный промышленный протокол Modbus и функционируют в режиме ведомого устройства (Modbus Slave). Шлюз NLS-Gateway выступает в роли ведущего устройства (Modbus Master), выполняя циклический опрос входных сигналов и передачу команд управления на выходы.

6.2.1. Предварительное конфигурирование модулей

Перед организацией опроса в IoT-платформе Home Assistant необходимо задать параметры связи на самом модуле:

Для модулей с поддержкой Modbus TCP (интерфейс Ethernet):

1. Подключитесь к встроенному веб-интерфейсу модуля через браузер (по заводскому IP-адресу, например, 192.168.0.1).
2. В разделе «Сетевые настройки / Modbus TCP» задайте:
 - IP-адрес и маску подсети: в соответствии с адресным пространством технологической сети;
 - TCP Port: порт службы Modbus (по умолчанию 502);
 - Unit ID (Slave Address): адрес устройства (по умолчанию 1).

Для модулей с поддержкой Modbus RTU (интерфейс RS-485):

1. Подключите модуль к ПК (через преобразователь интерфейсов USB/RS-485) и запустите сервисную утилиту для конфигурирования модулей ввода-вывода NLSconfig_v3 от компании RealLab.
2. Задайте параметры линии связи:
 - Режим протокола: выберите режим Modbus RTU (если по умолчанию установлен символьный протокол DCON);
 - Сетевой адрес (Unit ID): уникальный адрес в диапазоне от 1 до 247;
 - Параметры связи: скорость обмена (Baud Rate, например, 9600), биты данных (8), стоп-биты (1) и контроль чётности (Parity — None, Even или Odd).

Примечание: Подробное описание работы и настройки, а также карты регистров модулей ввода-вывода Modbus RTU/TCP приведены в эксплуатационной документации на странице конкретной модели на официальном сайте компании.

Информационный обмен с модулями Modbus RTU/TCP в IoT-платформе Home Assistant описан в подразделе 3.4.

Пример с описанием для работы с модулями ввода-вывода с поддержкой Modbus RTU/TCP в IoT-платформе Home Assistant в связке с Node-RED доступен для скачивания на странице шлюза на сайте компании RealLab.

6.3. Модули с поддержкой LoRaWAN

Беспроводная модули RealLab с поддержкой LoRaWAN (в частности, термоштанга TR-120091-L) предназначена для непрерывного контроля физических параметров (температуры, влажности) на распределенных объектах: в складах напольного хранения, силосах элеваторов, буртах, резервуарных парках и на открытых технологических площадках.

Телеметрия с датчиков передается по энергоэффективному радиоканалу LoRaWAN (диапазон RU864) на радиоконцентратор шлюза NLS-Gateway-LoRa. Встроенный сетевой сервер ChirpStack выполняет дешифрование данных и публикацию их в локальный брокер MQTT, откуда IoT-платформа Home Assistant считывает информацию для обработки, логического анализа и отображения на мнемосхемах оператора.

Подробный пример настройки обмена радиопакетами с термоштангой Reallab на шлюзе с Home Assistant доступен для скачивания на странице шлюза на сайте компании RealLab.

7. Диагностика и устранение неисправностей

В настоящем разделе описаны средства мониторинга состояния шлюза, способы просмотра системных журналов, а также приведены типовые неисправности и рекомендации по их устранению. Регулярный контроль загрузки ресурсов и анализ логов помогают своевременно выявлять отклонения в работе и предотвращать сбои.

7.1. Системный мониторинг ресурсов шлюза

Веб-конфигуратор RealLab! предоставляет наглядные инструменты для оценки текущей загрузки центрального процессора (CPU), использования оперативной памяти (RAM), температуры устройства и нагрузки от отдельных процессов (включая Home Assistant).

Примечание. Подробное описание работы веб-конфигуратора RealLab! приведено в руководстве пользователя по работе с ПО NLS-Gateway. Это руководство доступно для скачивания на странице шлюза на сайте компании RealLab.

7.1.1. Просмотр информации о системе

Для доступа к веб-конфигуратору шлюза, откройте веб-браузер и перейдите по адресу: `http://<IP-адрес_шлюза>:8000` (по умолчанию логин: `pi` и пароль: `12345678`). Перейдите на страницу «Инфо» (кнопка в шапке веб-интерфейса). Здесь отображаются:

- общая информация о шлюзе (имя хоста, версия ОС, время работы);
- нагрузка на каждое ядро CPU в процентах;
- использование оперативной памяти (в процентах и мегабайтах);
- занятость постоянной памяти (SD-карты);
- температура процессора (если поддерживается датчиком).

По умолчанию данные обновляются каждые 3 секунды (автообновление можно отключить).

Эти показатели позволяют быстро оценить, не перегружено ли шлюз, и принять решение о перезапуске служб или оптимизации проектов.

7.1.2. Управление процессами

Страница «Процессы» выводит список всех запущенных процессов с их идентификаторами (PID), потреблением CPU и памяти. Процессы отсортированы по убыванию нагрузки на процессор, что облегчает поиск «тяжёлых» задач.

На этой странице доступны следующие действия:

- завершение процесса;
- включение автообновления списка с интервалом 1, 3, 5 или 10 секунд;
- просмотр логов и управление службами – для этого нажмите кнопку «Логирование и управление службами» в верхней части страницы и выберите нужную службу из списка.

Мониторинг процессов особенно полезен при диагностике зависаний или аномально высокой загрузки, вызванной, например, нештатной работой скриптов Home Assistant или интенсивным сетевым обменом.

7.2. Просмотр системных журналов и диагностика ошибок

Большинство служб, установленных на шлюзе (ChirpStack, MQTT-брокер, Zigbee2MQTT, Home Assistant), ведут собственные журналы событий. Доступ к этим журналам организован единообразно через веб-конфигуратор RealLab!.

7.2.1. Общий подход к логированию

Для каждой службы в соответствующем разделе веб-конфигуратора предусмотрена кнопка «Логирование и управление службами» (или аналогичная). При её нажатии открывается диалоговое окно, в котором можно:

- выбрать конкретную службу из выпадающего списка;
- просмотреть логи (с возможностью ограничения количества строк от конца файла);
- скачать логи в файл;
- запустить, остановить, перезапустить службу;
- включить или отключить её автоматический запуск при загрузке шлюза.

Такой подход доступен для следующих служб:

- ChirpStack – логи концентратора, сетевого сервера и MQTT Forwarder;
- MQTT-брокер Mosquitto – логи брокера;
- Zigbee2MQTT – логи самого веб-интерфейса Z2M;
- Home Assistant – логи docker-контейнера;

7.2.2. Рекомендации по анализу логов

При поиске неисправностей обращайте внимание на следующее:

- ошибки (ERROR) – указывают на критические сбои, часто содержат код ошибки и описание причины;
- предупреждения (WARNING) – могут сигнализировать о некорректных настройках или приближении к лимитам ресурсов;
- сообщения отладки (DEBUG) – полезны для глубокого анализа, но их уровень должен быть включён в настройках соответствующей службы.

Если служба не запускается, сначала проверьте её статус через диалог управления службами, затем просмотрите логи – там обычно указывается конкретная причина отказа (например, неверный пароль в MQTT, отсутствие прав доступа к файлу, конфликт портов и т.д.).

Для удобства диагностики можно скачать логи и передать их технической поддержке RealLab!.

7.3. Типовые неисправности и методы их устранения

В табл. 5 приведены наиболее часто встречающиеся проблемы, их вероятные причины и рекомендуемые действия по восстановлению работоспособности.

Табл. 5. Интерфейсы Типовые неисправности шлюзов NLS-Gateway и способы их устранения

Неисправность	Возможная причина	Действия по восстановлению
Служба ChirpStack / ChirpStack Gateway Bridge не запущена	Сбой после обновления, нехватка памяти, повреждение конфигурации	Через веб-конфигуратор RealLab! в разделе управления службами-проверить статус и перезапустить службу. При необходимости просмотреть логи и исправить ошибки конфигурации.
Служба Home Assistant не запускается или работает нестабильно	Ошибка в конфигурационных файлах (YAML); повреждение базы данных (home-assistant_v2.db); некорректная работа сторонних интеграций или дополнений; недостаток оперативной памяти.	Просмотреть логи Home Assistant через веб-конфигуратор или веб-интерфейс. Попробовать запустить службу в безопасном режиме (Safe Mode). При повреждении базы данных временно переименовать её файл для автоматического создания нового или восстановить систему из резервной копии.
Нет связи с LoRaWAN-датчиками	Не совпадает Gateway ID в ChirpStack; неправильный частотный план; проблемы с радиомодулем	Проверить Gateway ID на вкладке LoRaWAN веб-конфигуратора и сравнить с идентификатором, добавленным в ChirpStack. Убедиться, что выбран правильный регион (частотный план) в параметрах концентратора. Проверить уровень сигнала и антенну.

7. Диагностика и устранение неисправностей

Неисправность	Возможная причина	Действия по восстановлению
Шлюз перезагружается/ зависает	Нестабильное питание, перегрев, ошибки файловой системы	Проверить температуру процессора (страница «Инфо»). Убедиться в исправности блока питания. Просмотреть системные логи через dmesg (по SSH) или через журналы служб. Выполнить перезагрузку.
Нет данных от Zigbee-устройств	Устройство не добавлено или потеряло связь; неправильно настроен MQTT-брокер; низкий уровень сигнала	Через веб-интерфейс Zigbee2MQTT убедиться, что устройство отображается в списке. Проверить качество связи (LQI) – при необходимости переместить шлюз или добавить маршрутизаторы. Проверить настройки подключения к MQTT.
Неверное время/дата в системе	Села батарейка RTC. Неверный часовой пояс	Проверить настройки часового пояса; Установить время вручную при необходимости через веб-интерфейс RealLab!
Конфликт IP-адресов	Два устройства в сети с одинаковым IP	Задать шлюзу статический IP через веб-интерфейс RealLab!

Лист регистрации изменений

Дата изменения	Описание изменения	Примечание