

Среда визуального программирования Node-RED

Программирование ПЛК RealLab в среде визуального программирования Node-RED

Программирование шлюзов RealLab в среде визуального программирования Node-RED

Руководство пользователя

© НИЛ АП, 2026

Версия 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	5
1.1. Введение	5
1.2. Общие сведения о Node-RED	5
1.3. Информация о шлюзах RealLab.....	6
1.3.1. Модельный ряд промышленных IoT шлюзов RealLab	6
1.3.2. Основные характеристики шлюза	6
1.3.3. Программное обеспечение и поддерживаемые протоколы	7
1.3.4. Интерфейсы шлюзов	8
1.3.5. Сетевые интерфейсы и настройка связи	10
2. Начало работы	12
2.1. Подключение к шлюзу NLS Gateway.....	12
2.2. Доступ к редактору среды Node-RED.....	13
2.3. Описание интерфейса редактора потоков	14
2.4. Управление проектами и отладка.....	22
2.4.1. Управление палитрой узлов.....	22
2.4.2. Экспорт и импорт потоков	26
2.4.3. Управление выполнением потоков	28
2.4.4. Отладка в среде Node-RED	31
2.4.5. Визуализация данных с помощью Node-RED Dashboard.....	33
3. Основной функционал работы Node RED.....	35
3.1. Беспроводные протоколы	35
Сбор данных по LoRa/ LoRaWAN	35
3.1.1. Bluetooth / BLE	37
3.2. Работа с протоколом MQTT.....	38
Подписка на топики (MQTT in).....	38
Публикация сообщений (MQTT Out).....	39
3.3. Промышленные протоколы связи	40
3.3.1. Интеграция с Modbus (RTU / TCP)	40
Программирование шлюзов RealLab	3
в среде визуального программирования Node-RED	

3.3.2. Работа шлюза как OPC UA-сервер	48
3.3.3. Работа шлюза как OPC UA-клиент	53
3.3.4. Настройка отправки уведомлений на Email.....	60
4. Работа с модулями ввода и вывода RealLab	62
4.1. Модули серии NLS xx Ethernet с поддержкой MQTT	62
4.2. Модули с поддержкой Modbus RTU/TCP	63
4.3. Модули с поддержкой LoRaWAN	65
5. Диагностика и устранение неисправностей	67
5.1. Системный мониторинг ресурсов шлюза.....	67
5.1.1. Просмотр информации о системе	67
5.1.2. Управление процессами.....	68
5.2. Просмотр системных журналов и диагностика ошибок	69
5.2.1. Общий подход к логированию	69
5.2.2. Рекомендации по анализу логов.....	69
5.3. Типовые неисправности и методы их устранения.....	70

1. Вводная часть

1.1. Введение

Настоящее руководство по программированию шлюзов серии NLS-Gateway в среде визуального программирования Node-RED содержит необходимую информацию для начала работы и освоения основных приёмов программирования. В нём приведено описание базового использования Node-RED в задачах промышленной автоматизации. Процесс ознакомления со средой описан применительно к шлюзам NLS-Gateway, однако значительная часть материала носит универсальный характер и может быть полезна при обучении работе с любыми устройствами, программируемыми в этой среде.

1.2. Общие сведения о Node-RED

Node-RED – это платформа визуального программирования с открытым исходным кодом, разработанная для упрощения создания приложений интернета вещей (IoT) и систем автоматизации. Благодаря low-code подходу, Node-RED позволяет пользователям проектировать логику приложений путём графического соединения функциональных блоков, называемых узлами (nodes), без необходимости написания большого объёма программного кода.

Node-RED состоит из двух ключевых компонентов:

- **Редактор потоков (Flow Editor)** – веб-интерфейс, в котором пользователь создаёт и настраивает логику приложения, соединяя узлы в последовательности, называемые потоками (flows).
- **Среда выполнения (Runtime)** – серверная часть, отвечающая за обработку и исполнение созданных потоков. В контексте данного руководства среда выполнения размещается непосредственно на промышленном шлюзе RealLab! серии NLS-Gateway под управлением ОС RealLab! Raspbian Linux arm64.

Основные преимущества применения Node-RED на шлюзе:

- **Гибкая фильтрация и парсинг.** Очистка и предобработка "сырых" данных прямо на шлюзе (Edge Computing) разгружает серверные ресурсы и экономит трафик сотовой связи.
- **Локальная автоматизация.** При потере связи с сервером шлюз продолжает выполнять локальные сценарии управления по RS-485, буферизируя критические данные.

- **Мультипротокольность.** Возможность объединить в одну логическую схему Modbus RTU/TCP, LoRaWAN, MQTT, HTTP, OPC UA и отправлять отчеты на электронную почту или СУБД.
- **Визуальное программирование.** Инженеры КИПиА могут быстро вносить изменения в логику работы шлюза без написания громоздкого кода на C/C++ или Python.

Благодаря своей гибкости и простоте освоения, Node-RED становится всё более популярным инструментом не только среди разработчиков, но и среди инженеров-автоматчиков, стремящихся быстро реализовать надёжные и масштабируемые решения для промышленной автоматизации.

1.3. Информация о шлюзах RealLab

1.3.1. 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 Node-RED (суффикс -NR):

- NLS-Gateway-LoRa-2RS-NR
- NLS-Gateway-LoRa-2RS- PoE-NR
- NLS-Gateway-Zigbee-2RS-NR
- NLS-Gateway-Zigbee-2RS-PoE-NR

Примечание: кроме перечисленных моделей, есть модификации с суффиксами -CDS (CODESYS) и -HA (Home Assistant).

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).

1.3.2. 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 , WiFi, Bluetooth.
- **Питание:** 10–30 В постоянного тока (диапазон может варьироваться в зависимости от модели).
- **Рабочая температура:** от –20 °С до +70 °С.
- **Крепление:** DIN-рейка.

1.3.3. 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.

Программное обеспечение и средства разработки

Шлюзы предоставляют следующие программные средства для разработки и настройки:

- **Веб-конфигуратор RealLab** – для настройки и диагностики 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 шлюзов 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).
- «-НА» – 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.

Поддерживаемые промышленные протоколы

Набор промышленных протоколов зависит от наличия интерфейса CAN в конкретной модели:

- 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 «1RS-1CAN») дополнительно поддерживают протокол CANOpen.
- Модели без CAN (с маркировкой «2RS») не поддерживают 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.

1.3.4. Интерфейсы 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 объемом 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.
Zigbee 3.0	Встроенный модуль (для Zigbee-моделей)	Диапазон частот 2405–2480 МГц, 16 каналов.

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 временной метки событий.

1.3.5. Сетевые интерфейсы и настройка связи

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).

1.1.4.1. Интерфейс 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, подробную информацию смотрите в **руководстве пользователя по работе с ПО шлюзов RealLab! серии NLS-Gateway**, в пункте 2.10 «**Настройка параметров сети**», руководство доступно на странице шлюза на сайте компании RealLab.

1.1.4.1. Интерфейс Wi-Fi

Режим «Точка доступа» (Access Point)

Точка доступа Wi-Fi. 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, подробную информацию смотрите в пункте 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 (IP-адрес-шлюза:8000, Login: pi, password:12345678).
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.

Подробную информацию 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.

1.1.4.3. 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

2.1. Подключение к шлюзу NLS Gateway

Для начала работы со шлюзом серии NLS-Gateway под управлением среды исполнения Node-RED, необходимо выполнять следующие действия:

1. Закрепите шлюз NLS-Gateway на DIN-рейке в месте, обеспечивающем оптимальный радиоприем.
2. Установите внешнюю антенну LoRaWAN на ВЧ-разъем типа SMA шлюза, а также антенны Wi-Fi (при использовании беспроводной сети).
3. Подключите источник питания к шлюзу и дождитесь полной загрузки операционной системы. После включения шлюза 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) (см. рис. 2.1).

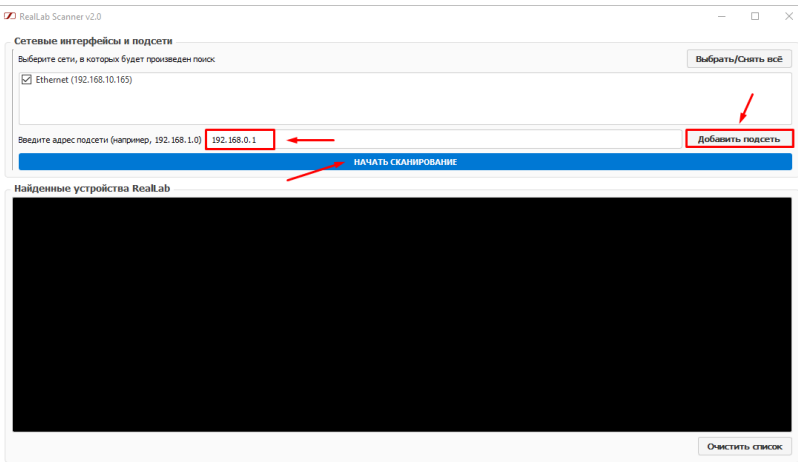


Рис. 2.1. Общий вид утилиты RealLab! Scanner v2.0

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

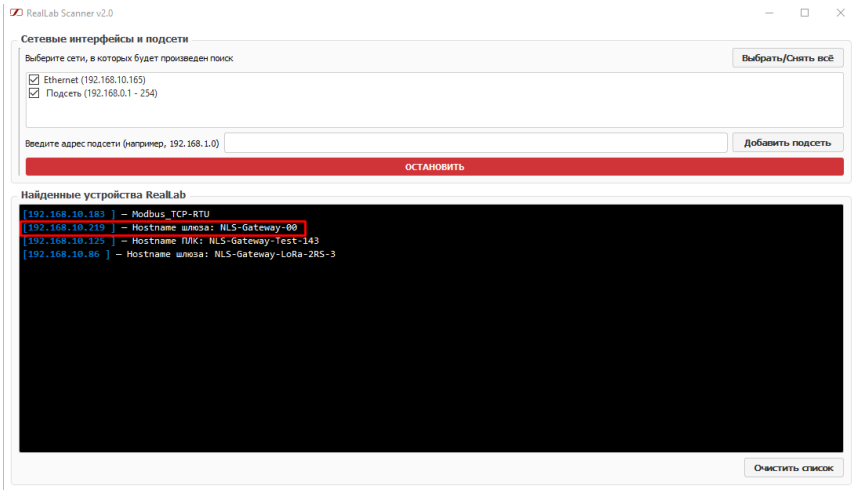


Рис. 2.2. Общий вид найденных устройств в окне результатов

2.2. Доступ к редактору среды Node-RED.

Среда визуального программирования Node-RED предоставляет доступ к разработке через встроенный веб-интерфейс. Для подключения к редактору выполните следующие действия:

1. Убедитесь, что шлюз включен, загружен и подключен к локальной сети, а ПК оператора находится в сети.
2. Откройте любой современный веб-браузер (рекомендуются Google Chrome, Яндекс.Браузер или Mozilla Firefox).
3. В адресной строке браузера введите сетевой адрес шлюза с указанием стандартного порта среды Node-RED — 1880:

Ошибка! Недопустимый объект гиперссылки.

(где <IP-адрес_шлюза> — адрес, определенный на этапе подключения согласно п. 2.1, например: <http://192.168.10.229:1880>).

4. При включенной функции аутентификации введите учетные данные (логин и пароль) администратора.

После успешной авторизации откроется главное окно редактора потоков (рис. 2.3).

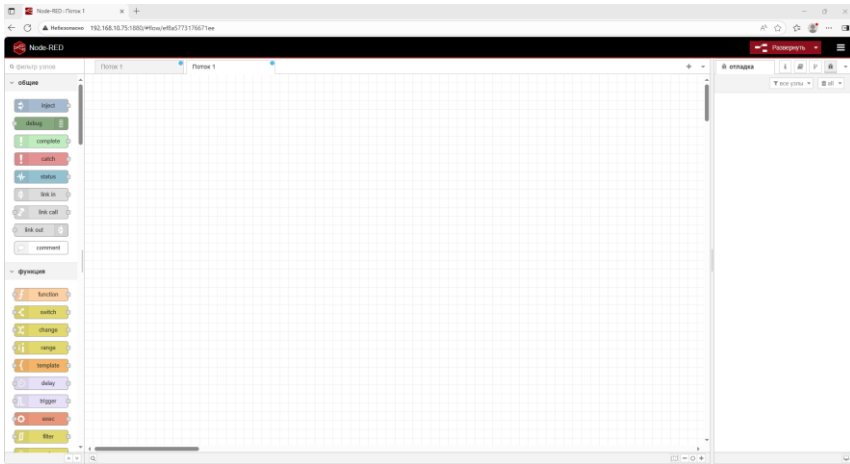


Рис. 2.3. Главное окно редактора потоков Node-RED

2.3. Описание интерфейса редактора потоков

Рассмотрим основные элементы интерфейса редактора потоков (см. рис. 2.4):

- Узлы (Nodes);
- Поток (Flow);
- Развертывание (Deploy);
- Боковая панель.

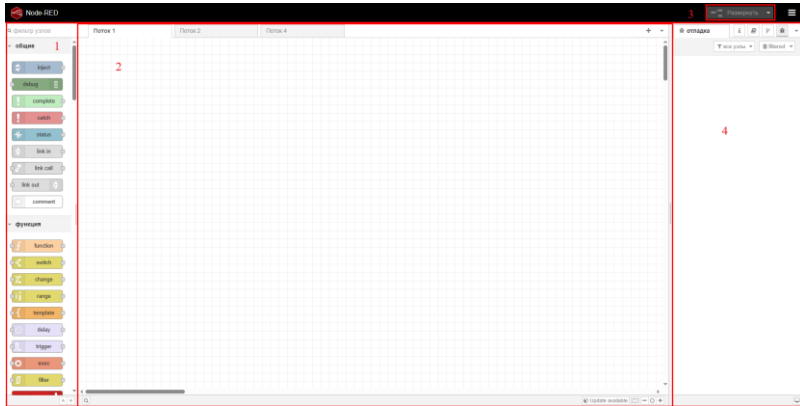


Рис. 2.4. Основные элементы интерфейса

1. Узлы (Nodes) – представляют собой боковую панель, содержащую все установленные и доступные для использования узлы.

Расположенная в верхней части палитры строка поиска позволяет быстро находить узлы по их названию.

Палитра разделена на несколько категорий (общие, функции и т.д.), каждая из которых содержит коллекции узлов. Когда устанавливается сторонний узел, он может создать новую категорию. Вы можете свернуть или развернуть категории, нажав на конкретную категорию.

В нижней части палитры вы найдете два значка со стрелками. Нажатие на первую иконку со стрелкой вверх сворачивает все категории, нажатие на значок со стрелкой вниз по умолчанию развернет все категории обратно.

При нажатии на кнопку посередине боковая панель палитры скрывается. Чтобы отобразить ее снова, нажмите на кнопку еще раз (см. рис. 2.5).



Рис. 2.5. Элементы управления узлами

2. Поток (Flow) – рабочая область, где потоки (группы узлов) развиваются путем перетаскивания узлов из палитры и связывания их вместе (см. рис. 2.6). При добавлении новой вкладки потока создается новое рабочее пространство.

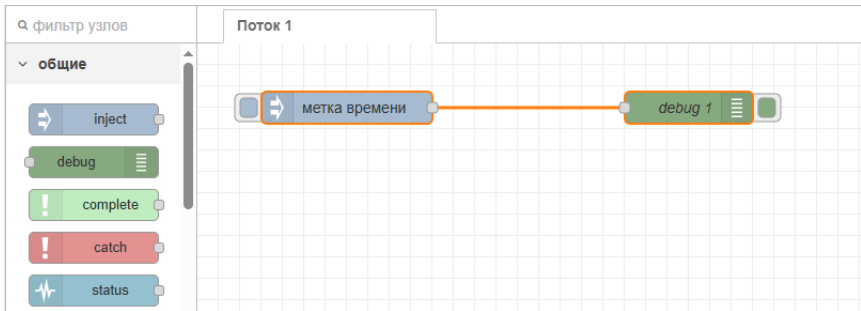


Рис. 2.6. Рабочая область (поток)

Чтобы создать родительский поток, нажмите на значок «+» в правом верхнем углу (см. рис. 2.7), в противном случае вы можете использовать опцию добавления потока в главное меню (см. рис. 2.8).

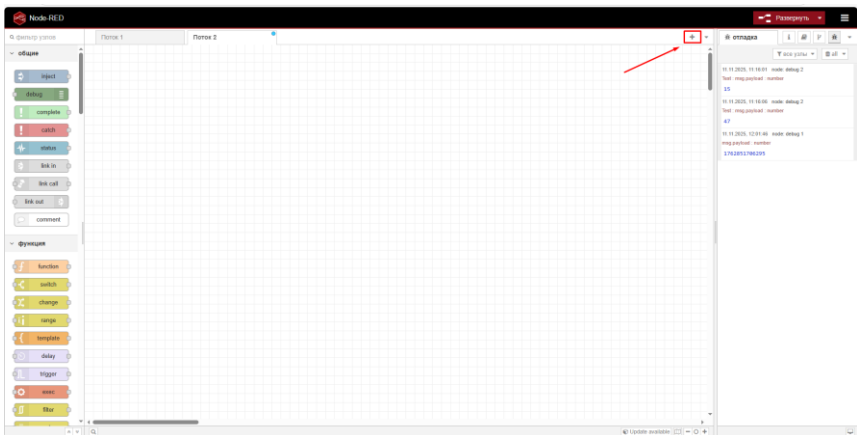


Рис. 2.7. Создание потока

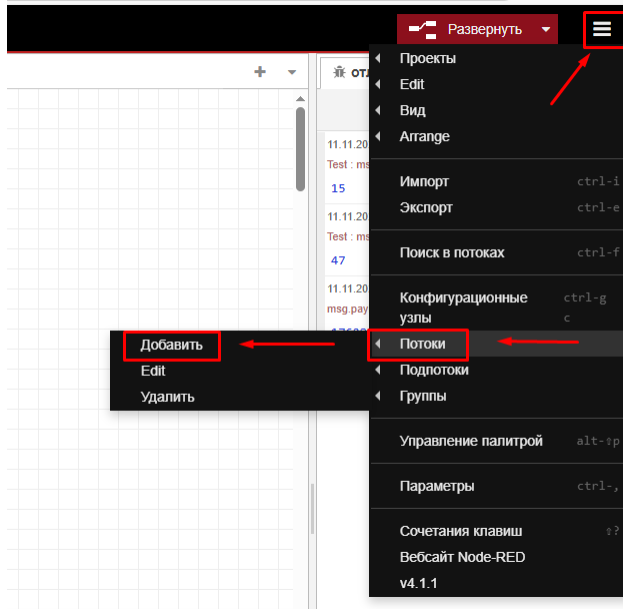


Рис. 2.8. Создание потока через главное меню

Чтобы включить или отключить поток, дважды щелкните вкладку потока. Нажмите внизу на кнопку «Отключить» или «Включить», если она уже отключена (см. рис. 2.9).

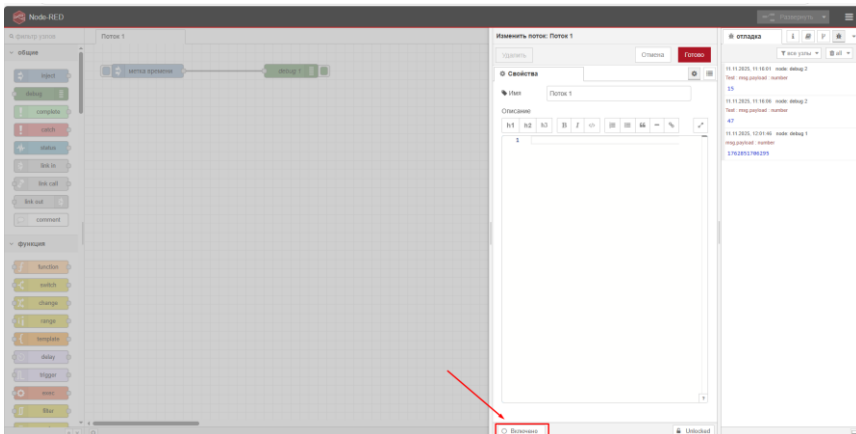


Рис. 2.9. Включение/отключение потока

Подпоток в Node-RED – это набор узлов, которые свернуты в отдельный узел в рабочей области. Чтобы создать подпоток используется опция добавления подпотока в главное меню (см. рис. 2.10). Созданный подпоток можно использовать как обычный узел в потоках.

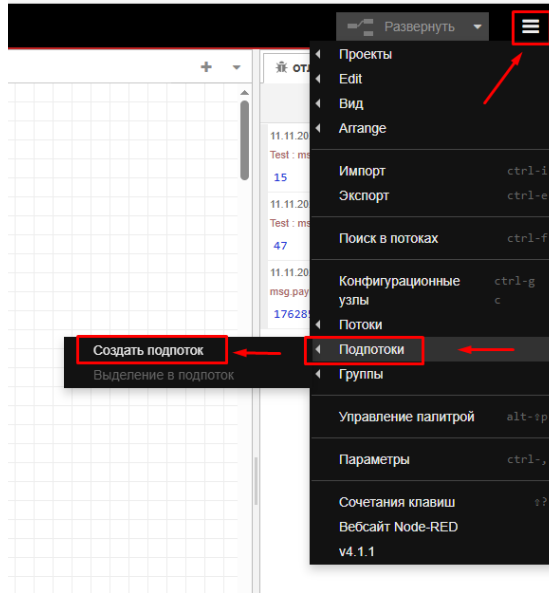


Рис. 2.10. Создание подпотока через главное меню

3. Развертывание (Deploy) – в верхней правой части расположена красная кнопка развертывания. При нажатии на нее выполняются все потоки внутри экземпляра. Красный цвет указывает на ожидающие изменения для развертывания, а серый – на отсутствие ожидающих изменений (см. рис. 2.11).

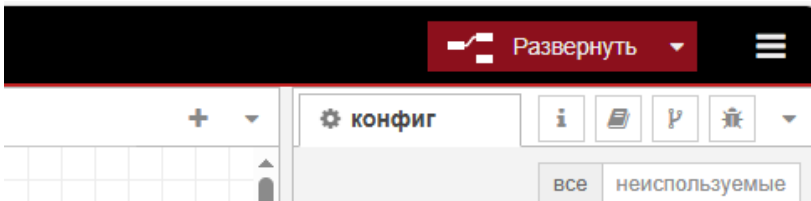


Рис. 2.11. Развертывание потоков

Чтобы развернуть только измененные потоки, или узлы, или остановить/перезапустить выполнение потока, нажмите значок развертывания на кнопке

развертывания. Выберите предпочтительный вариант и нажмите соответствующую кнопку развертывания (см. рис. 2.12).

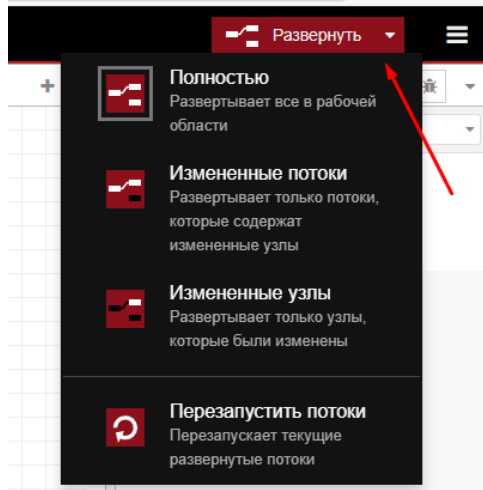


Рис. 2.12. Выбор варианта развертывания потоков

4. Боковая панель – предоставляет дополнительные контекстно-зависимые параметры и информацию в зависимости от выбранного узла или рабочей области. Инструменты, доступные на боковой панели, называются панелями (см. рис. 2.13).

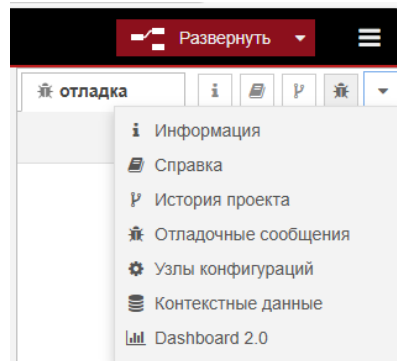


Рис. 2.13. Боковая панель

Боковая панель Node-RED по умолчанию включает в себя следующие панели:

Информация – показывает информацию обо всех потоках, присутствующих в них узлах, подпотоках, а также узлах глобальной конфигурации в древовидной структуре.

Свойства элемента – включает в себя (см. рис. 2.14):

- справку, клик по ней приведет к переходу на панель помощи и покажет предоставленный автором файл readme для выбранного элемента;
- копирование URL выбранного элемента, вы можете использовать эту функцию, когда хотите указать или обсудить конкретный элемент в рабочем пространстве с членом вашей команды;
- позволяет найти этот элемент в рабочей области.

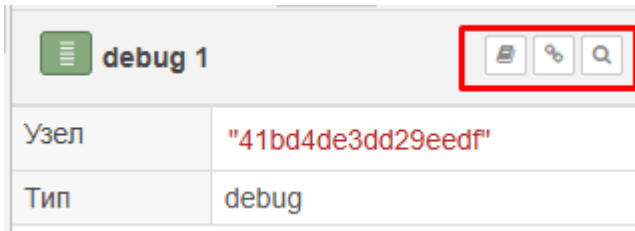


Рис. 2.14. Свойства элемента

Справка (Help) – представляет собой древовидную структуру и позволяет просматривать документацию.

Отладка (Debug) – отображает сообщения, выводимые узлом отладки, эта панель помогает легко отладить ваше приложение. Включает в себя опцию фильтра, позволяющую отображать информацию по:

- всем узлам - при выборе этой опции будут отображаться сообщения, напечатанные всеми узлами;
- выбранным узлам - позволяет выбрать конкретные узлы отладки, сообщения которых вы хотите видеть на панели отладки;
- текущему потоку - будут печататься только сообщения, напечатанные узлами, которые присутствуют в текущем потоке.

Примечание: Боковая панель "Отладка" может отображать только 100 самых последних сообщений. Если на боковой панели в данный момент отображается отфильтрованный список сообщений, скрытые сообщения по-прежнему учитываются в лимите в 100 сообщений.

Контекстные данные – на этой панели отображаются контекстные переменные по их области. Каждая переменная включает в себя такие сведения, как дата и время, ее хранилище (в памяти или постоянное), а также имя и значение.

Узлы конфигурации (Config nodes) – значок шестеренки показывает информацию об узлах конфигурации. Узлы конфигурации не отображаются в основном рабочем пространстве, и это специальные узлы, которые содержат повторно используемые конфигурации, которые могут совместно использоваться несколькими узлами в потоке.

2.4. Управление проектами и отладка

2.4.1. Управление палитрой узлов

Узлы **Node-RED** устанавливаются через Менеджер палитр – инструмент, позволяющий пользователям управлять узлами, доступными для использования в **Node-RED**. Менеджер палитр показывает все установленные узлы и позволяет пользователям устанавливать, обновлять и удалять их при необходимости. Для установки необходимого пакета узлов:

1. Откройте редактор **FlowFuse Node-RED**.
2. Нажмите на меню в правом верхнем углу и выберите «**Управление палитрой**» (см. рис. 2.15).

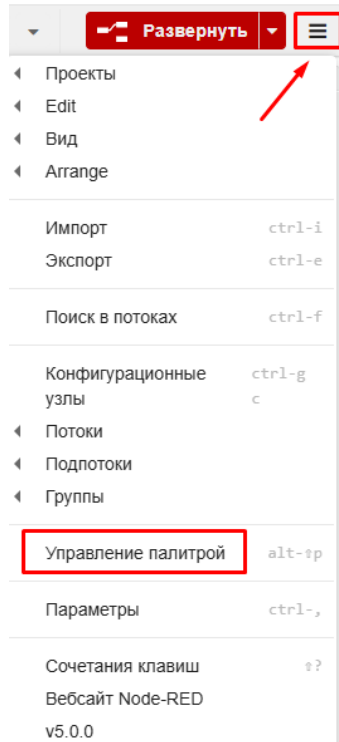


Рис. 2.15. Раздел управление палитрой в главном меню

3. Перейдите на вкладку «Установить» и выполните поиск по запросу, например, `node-red-contrib-usbcamera` (см. рис. 2.16):

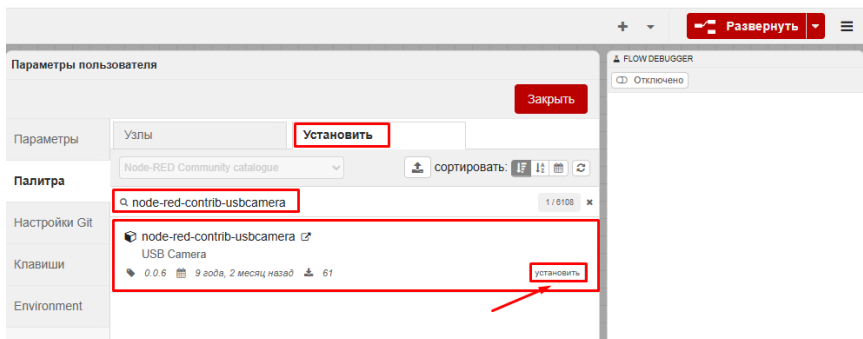


Рис. 2.16. Установка узлов для работы с usbcamera

4. Нажмите на кнопку «Установить» во всплывающем окне (см. рис. 2.17).

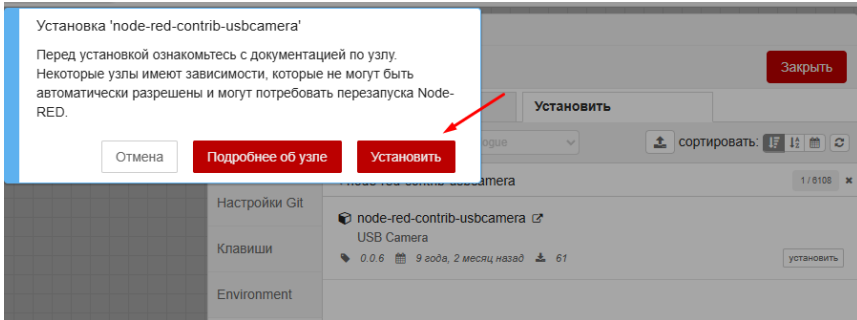


Рис. 2.17. Установка узлов для работы с usbcamera

После установки новые узлы отобразятся на палитре. Аналогичным образом устанавливаются пакеты узлов, необходимые для работы проекта.

2.4.1.1 Просмотр списка установленных пакетов

Чтобы посмотреть список установленных пакетов необходимо:

1. ПКМ кликнуть на рабочей области, в открывающемся списке выбрать пункт **Show action list** (см. рис. 2.18).

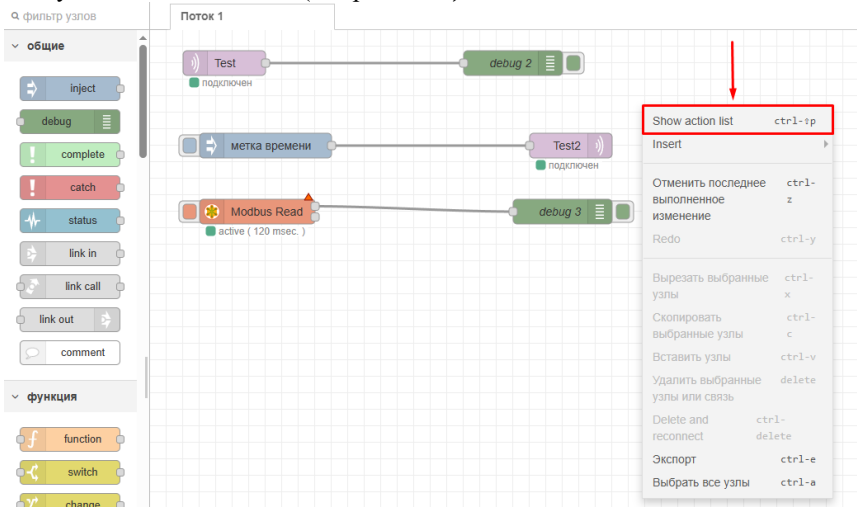


Рис. 2.18. Открытие списка Show action list

2. Далее в поисковой строке необходимо ввести **System Info** (см. рис. 2.19).

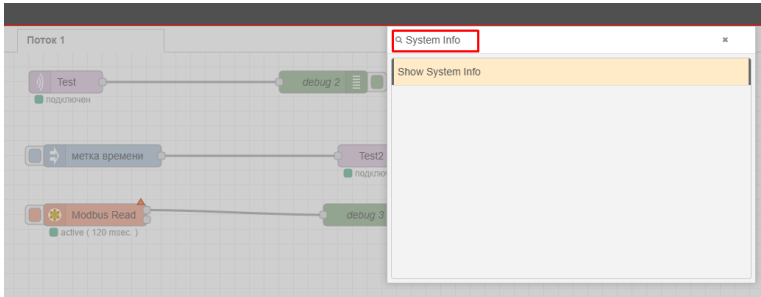


Рис. 2.19. Открытие списка Show action list

3. В открывшемся окне перейти на вкладку Visual. Найти «modules», в котором будет перечень всех установленных пакетов (см. рис. 2.20).

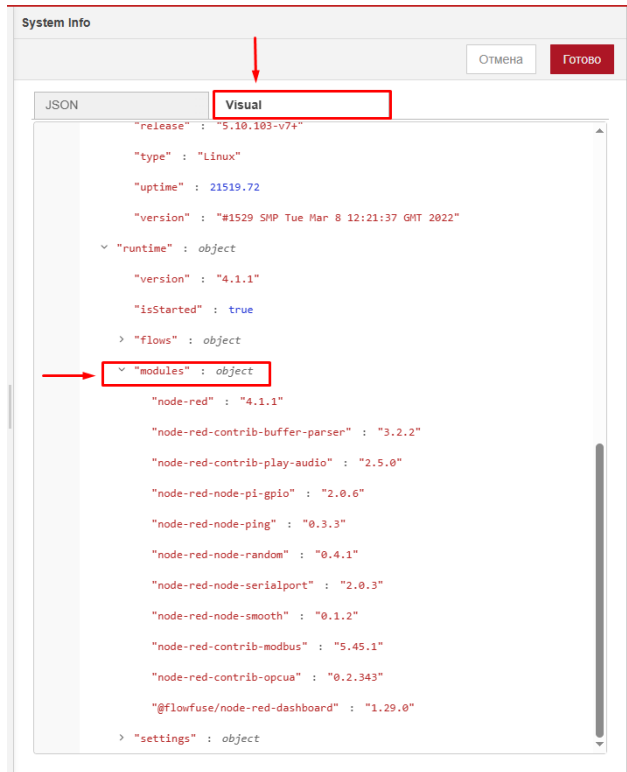


Рис. 2.20. Отображение перечня всех установленных пакетов

2.4.2. Экспорт и импорт потоков

Среда Node-RED поддерживает функции экспорта, импорта и группировки потоков. Это позволяет создавать резервные копии проектов, обмениваться конфигурациями с другими пользователями и структурировать связанные узлы для повышения удобства администрирования.

Для сохранения конфигурации ваших узлов откройте главное меню в правом верхнем углу интерфейса разработки и выберите пункт «Export» (Экспорт) (см. рис. 2.21).

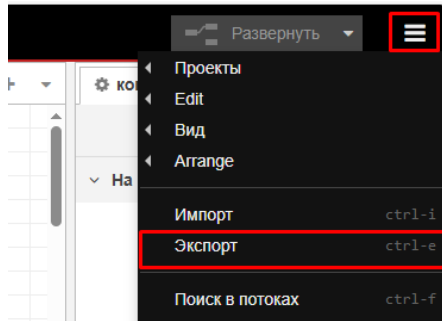


Рис. 2.21. Открытие «Экспорт узлов»

В открывшемся окне экспорта (см. рис. 2.22) выберите область загрузки: выделенные узлы, текущий поток или весь проект. Конфигурацию можно сохранить в текстовом формате JSON (скопировав ее в буфер обмена) либо выгрузить в виде отдельного файла на компьютер

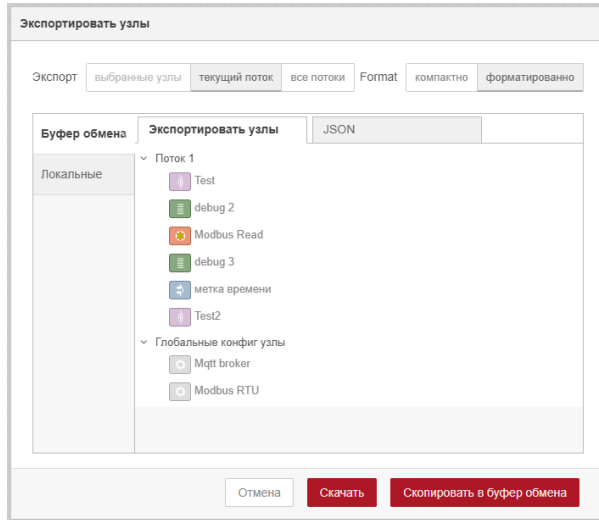


Рис. 2.22. Экспортировать узлы

Для загрузки ранее сохраненной конфигурации откройте главное меню, выберите пункт «**Import**» (Импорт) (см. рис. 2.23), вставьте текстовую строку в формате JSON в соответствующее поле или выберите JSON-файл на вашем компьютере (см. рис. 2.24).

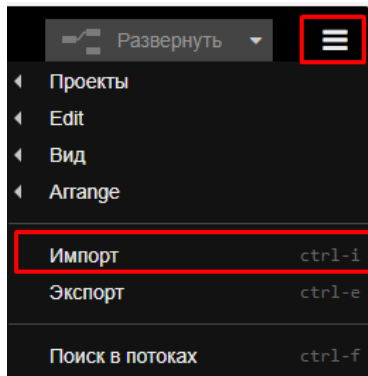


Рис. 2.23. Открытие «Импорт узлов»

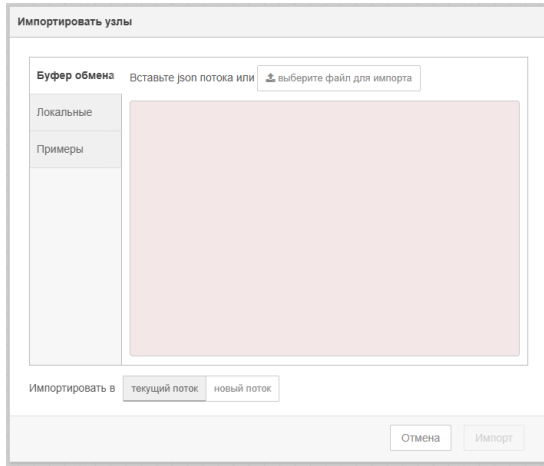


Рис. 2.24. Импортировать узлы

2.4.3. Управление выполнением потоков

2.4.3.1 Запуск и остановка потоков через среду разработки

Для добавления опции Start/Stop в выпадающем меню кнопки Deploy (Развернуть) необходимо изменить настройки **false** на **true** в разделе **runtimeState**.

Для этого выполните следующие действия:

1. Подключитесь к ПЛК либо напрямую, либо удалённо по протоколу SSH.
2. Откройте терминал и с помощью текстового редактора nano отредактируйте файл:
`$sudo nano /home/pi/.node-red/settings.js`
3. Найдите в файле параметр `runtimeState` и установите его значение в `true`.
4. Сохраните изменения (в nano — нажмите Ctrl+O, затем Enter) и выйдите из редактора (Ctrl+X).
5. Перезапустите среду выполнения Node-RED или перезагрузите ПЛК, чтобы изменения вступили в силу.

Чтобы убедиться, что настройка применена, выполните следующее:

1. Щёлкните правой кнопкой мыши на рабочей области **Node-RED**.
2. В открывшемся контекстном меню выберите пункт **Show action list**.

3. В поисковой строке введите **Show System Info** и откройте соответствующий элемент.
4. Перейдите на вкладку **Visual**, где в разделе **runtimeState** вы сможете увидеть текущее состояние (см. рис. 2.25).

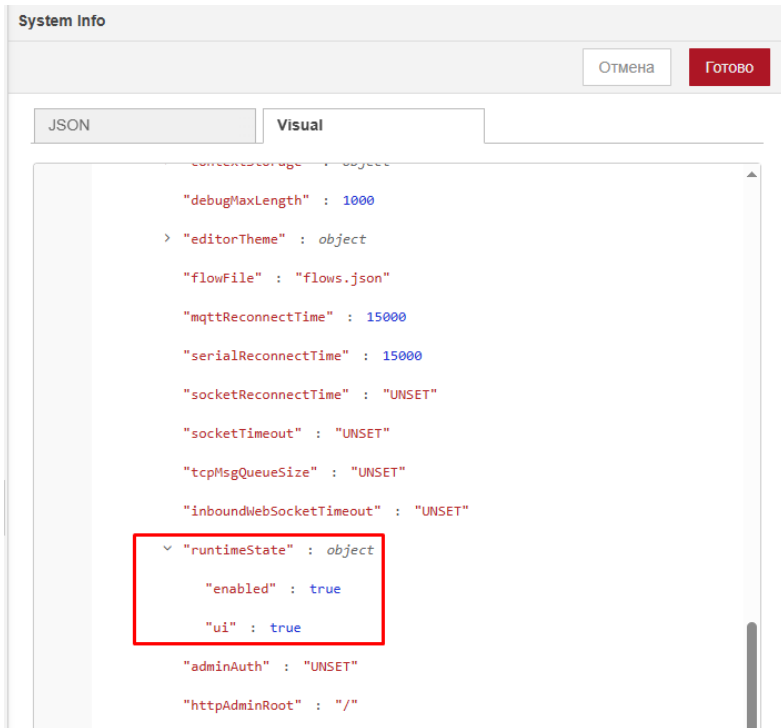


Рис. 2.25. Информации о настройках runtimeState

В результате станет доступна опции Start/Stop в выпадающем меню кнопки Deploy (Развернуть) (см. рис. 2.26).

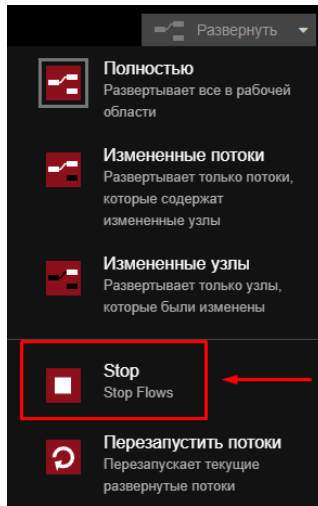


Рис. 2.26. Выбор варианта развертывания потоков

2.4.3.2 Перезагрузка среды исполнения Node-RED

Перезагрузка Node-RED через среду разработки

Перезагрузку runtime в Node-RED через среду **разработки** можно реализовать с помощью специальных пакетов узлов, которые позволяют запускать команды перезагрузки при определенных условиях (или по расписанию) и сохранять данные после перезагрузки Node-RED.

Пакет **node-red-contrib-rpi-shutdown** содержит узлы «Shutdown RPi» и «Restart RPi», которые запускают команду перезагрузки при любом условии или по расписанию (см. рис. 2.27 **Ошибка! Источник ссылки не найден.**).

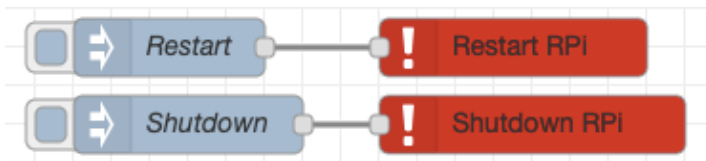


Рис. 2.27. Узлы для перезагрузки Node-RED

Перезапустить узел: если этот узел сработает, шлюз будет перезагружен.

Перезагрузка через веб-конфигуратор RealLab!

В случае необходимости перезапуска службы Node-RED через веб-конфигуратор RealLab!:

1. Откройте веб-конфигуратор RealLab!, введя в адресную строку браузера сетевой адрес шлюза с обязательным указанием порта 8000 (например, <http://192.168.1.100:8000>). Авторизуйтесь, используя учётные данные по умолчанию: логин `pi`, пароль `12345678`.
2. После авторизации перейдите в раздел «Интеграции» и выберите вкладку «Node-RED» (см. рис. 2.28);

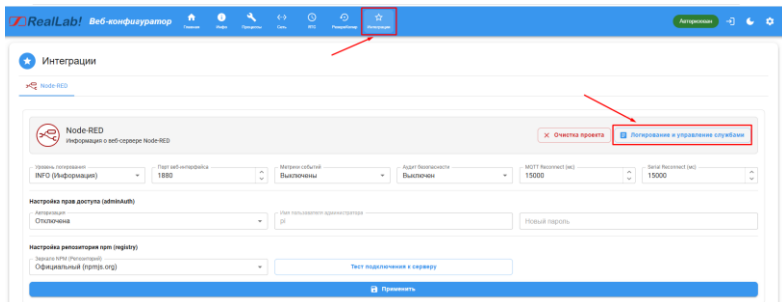


Рис. 2.28. Открытие вкладки «Логирование и управление службами»

3. Нажмите кнопку «Логирование и управление службами», в открывшейся панели управления службой `nodered.service` нажмите кнопку «Перезапустить» (Restart).

2.4.4. Отладка в среде Node-RED

2.4.4.1. Установка и включение отладчика

Чтобы установить отладчик Node-RED нажмите на значок меню в правом верхнем углу, выберите «Управление палитрой» и перейдите на вкладку «Установить». Найдите **node-red-debugger**.

Нажмите «Установить», чтобы добавить пакет (см. рис. 2.29).

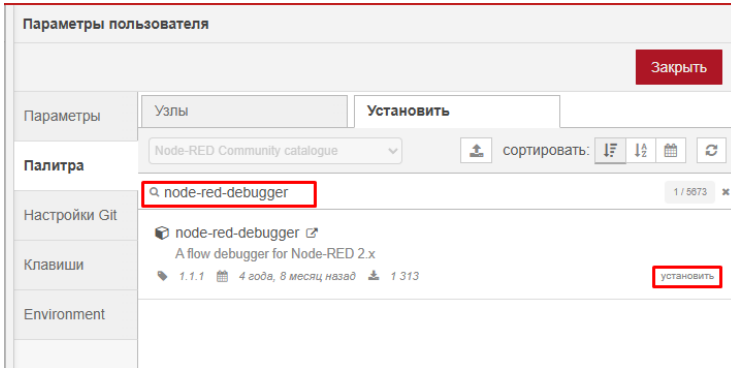


Рис. 2.29. Установка отладчика

После установки откройте вкладку «Отладка (debugger)» на боковой панели, щелкнув значок сворачиваемой стрелки на правой боковой панели и выберите «Отладчик потока» (Flow Debugger) (см. рис. 2.31).

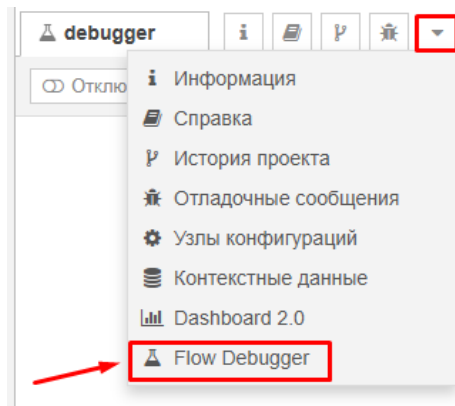


Рис. 2.30. Открытие «Отладчик потока»

На новой вкладке «Отладчик» переключите переключатель в верхнем левом углу боковой панели, чтобы включить отладчик. По умолчанию он отключен, поэтому включите его, прежде чем продолжить (см. рис. 2.31).

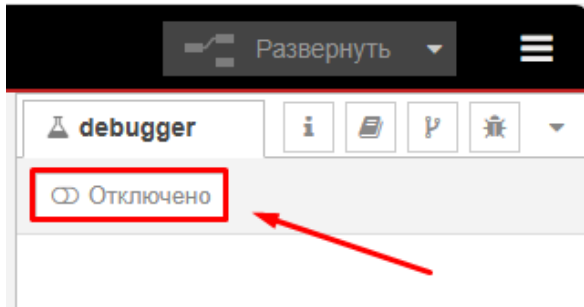


Рис. 2.31. Включение «Отладчик потока»

2.4.5. Визуализация данных с помощью Node-RED Dashboard

Node-RED Dashboard – это набор узлов и компонентов пользовательского интерфейса, которые позволяют создавать веб-панели в Node-RED. Он предоставляет виджеты, такие как кнопки, диаграммы, датчики и текстовые поля для отображения и взаимодействия с данными из ваших потоков в режиме реального времени.

Для установки пакета необходимо:

1. В меню в правом верхнем углу выбрать «**Управление палитрой**», перейти на вкладку «Установить» и выполнить поиск по запросу: `@flowfuse/node-red-dashboard`
2. После установки откройте вкладку на правой боковой панели и выберите «Dashboard 2.0» (см. рис. 2.32).

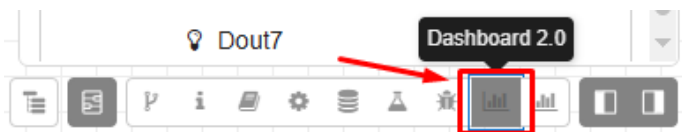


Рис. 2.32. Открытие «Dashboard»

Dashboard добавляет отдельную боковую панель в Node-RED для централизованного просмотра ваших страниц, групп и виджетов. Отсюда вы можете добавлять новые страницы и группы, менять существующие настройки и перестраивать контент по своему вкусу (см. рис. 2.33).

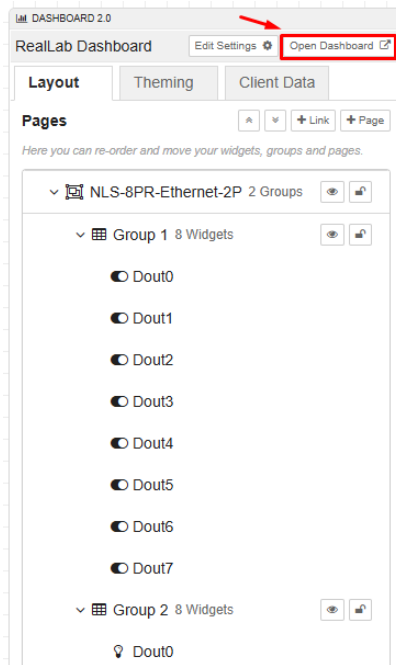


Рис. 2.33. Боковая панель «Dashboard»

3. Перетащите необходимые узлы из палитры в рабочую область, нажмите кнопку «Развернуть», а затем на кнопку «Open Dashboard» (см. рис. 2.34). **Ошибка! Источник ссылки не найден.**

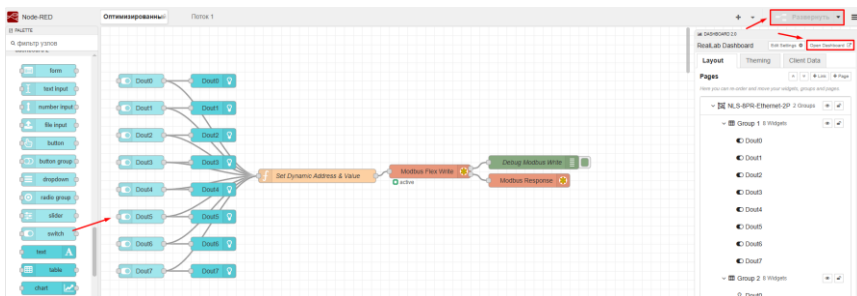


Рис. 2.34. Работа с узлами палитры «Dashboard»

Dashboard автоматически создаст новую страницу, в которой будут располагаться выбранные узлы (см. рис. 2.35).

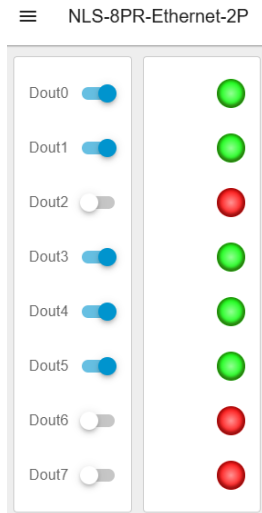


Рис. 2.35. Окно визуализации

При нажатии на переключатель лампочка будет менять цвет (зеленый/ красный) и соответствующий индикатор будет загораться (выключаться) на модуле вывода.

3. Основной функционал работы Node RED

При разработке в среде Node-RED ключевое значение имеет организация надежного информационного обмена между устройствами. Взаимодействие обеспечивается поддержкой широкого набора сетевых протоколов, таких как MQTT, HTTP, WebSockets, а также промышленных стандартов Modbus и OPC UA.

3.1. Беспроводные протоколы

Сбор данных по LoRa/ LoRaWAN

Архитектура приёма данных с LoRaWAN-устройств строится по классической многоуровневой схеме:

3. Основной функционал работы Node RED

1. **Конечное устройство** периодически (с заданным интервалом) опрашивает сенсоры и передает по радиоканалу LoRa радиопакеты на шлюз, содержащие идентификатор устройства, например, значения температуры, влажности, уровень заряда батареи и служебную информацию.
2. **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.
3. Сетевой сервер ChirpStack публикует данные в MQTT-брокер в топики.
4. Node-RED выступает в роли потребителя данных, подписываясь на топики брокера MQTT, парсит JSON-сообщения и передаёт обработанные данные во внешние системы (SCADA, БД, облако).
5. Пользователь через любой современный браузер обращается к Dashboard по HTTP и видит актуальную информацию в реальном времени.

На рис. 3.1 приведена обобщённая схема информационных потоков в системе термометрии с LoRa датчиками.

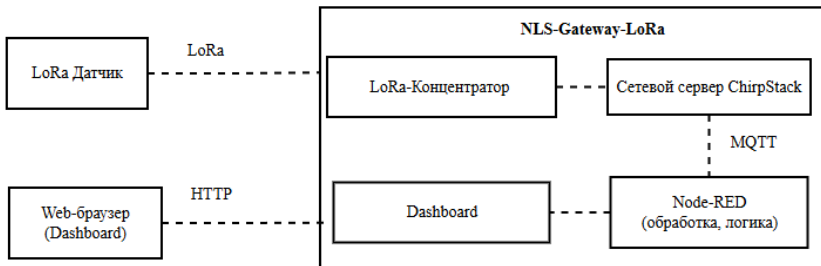


Рис. 3.1. Архитектура LoRa интеграции в шлюзе

Приём и декодирование пакетов от LoRa-датчиков

Для подключения датчика и приёма его телеметрии в среде Node-RED необходимо выполнить следующие шаги:

1. Настройка в ChirpStack:

- В веб-интерфейсе сетевого сервера ChirpStack создать профиль устройства (Device Profile) и зарегистрировать приложение (Application).

3. Основной функционал работы Node RED

- Добавить конечное устройство, указав идентификаторы и ключи аутентификации (DevEUI, AppKey / AppSKey, NwkSKey в зависимости от режима активации ОТАО или ABP).
- В свойствах профиля устройства добавить JavaScript-код декодера (парсера) для автоматического декодирования (Payload Codec).

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

2. Настройка приёма в Node-RED и обработка данных:

- В редакторе Node-RED, создать узел mqtt in.
- В конфигурации узла указать IP-адрес шлюза (или localhost, если брокер локальный), порт 1883 и параметры авторизации.
- Настроить подписку на топик событий приложения:

application/{application_id}/device/{dev_eui}/event/up

Подробнее описание о настройке узлов MQTT и обработки данных см. подраздел 3.2.

Примечание: Готовый пример потока Node-RED и подробное описание интеграции термостанга TR120091_L по протоколу LoRaWAN доступны для скачивания на странице шлюза на официальном сайте компании.

3.1.1. Bluetooth / BLE

Node-RED позволяет настроить связь Bluetooth / BLE с беспроводными датчиками, метками и периферийными устройствами. Для этого потребуется установить необходимый пакет узлов: **node-red-contrib-ble-sense**.

Ниже приведён пример потока, который обеспечивает подписку на все характеристики (см. рис. 3.2).

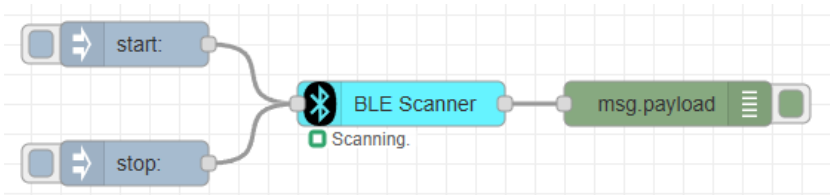


Рис. 3.2. Настройка Bluetooth соединения

3.2. Работа с протоколом MQTT

Подписка на топики (MQTT in)

1. Для приема сообщений из брокера используется узел MQTT In. Добавьте узел debug (для отображения, получаемого узлом сообщения). Подключите его к узлу MQTT In (см. рис. 3.3).

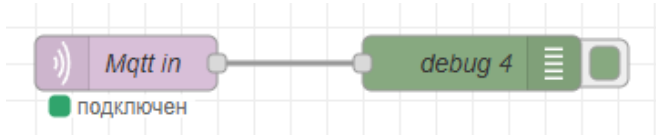


Рис. 3.3. Прием сообщений из брокера

2. Дважды кликните мышкой по узлу MQTT In, чтобы открыть окно настроек (см. рис. 3.3).
3. Задайте сервер (адрес брокера MQTT), тему подписки, значения параметров QoS и выхода (см. рис. 3.4). Для получения данных с датчиков задайте тему в формате:

application/<ApplicationID>/device/<DevEUI>/event/up

The screenshot shows the 'Изменить узел mqtt in' (Edit MQTT In node) configuration window. The window has a title bar and buttons for 'Удалить' (Delete), 'Отмена' (Cancel), and 'Готово' (Done). Below the buttons is a 'Свойства' (Properties) tab. The configuration fields are as follows:

- Сервер** (Server): A dropdown menu showing '192.168.0.87:1883'. A red box highlights this field, and a red arrow points to the edit icon (pencil) next to it.
- Action**: A dropdown menu showing 'Subscribe to single topic'.
- Тема** (Topic): A text input field containing 'application/9ee8f8ed-3a7f-479d-9189-1d01c3628ecc/device/'. A red box highlights this field.
- QoS**: A dropdown menu showing '2'.
- Flags**: Two checkboxes: 'Do not receive messages published by this client' (unchecked) and 'Keep retain flag of original publish' (unchecked).
- Retained message handling**: A dropdown menu showing 'Send retained messages'.
- Выход** (Output): A dropdown menu showing 'автоопределение (разобрать объект JSON, строка или...)'.
- Имя** (Name): A text input field containing 'Mqtt in'.

Рис. 3.4. Окно настроек узла MQTT In

3. Основной функционал работы Node RED

- Для узла **MQTT In** необходимо настроить соединение с брокером MQTT, которое настраивается нажатием на значок карандаша (см. рис. 3.4).

Стандартный порт MQTT-брокера для входящих TCP-соединений – 1883 (см. рис. 3.5). При использовании защищенного SSL-подключения используется порт 8883.

Изменить узел mqtt in > Изменить узел mqtt-broker

Удалить Отмена Обновить

Свойства

Имя Имя

Соединение Безопасность Сообщения

Сервер 192.168.0.87 Порт 1883

☒ Connect automatically
☐ TLS

Protocol MQTT V5

ID клиента Оставьте пустым для автоматически сгенерированного

Keep-alive время (сек) 60

Session ☒ Use clean start
Session Expiry (secs)

User Properties none

Рис. 3.5. Настройка соединения с брокером MQTT

- Разверните поток «Развернуть» (Deploy). Узел MQTT In автоматически переподключается в случае потери соединения с брокером.

Публикация сообщений (MQTT Out)

Для публикаций сообщений в MQTT брокер используется узел MQTT Out.

- Добавьте узел inject (для отправки текстового сообщения). Подключите его к узлу MQTT Out (см. рис. 3.6).

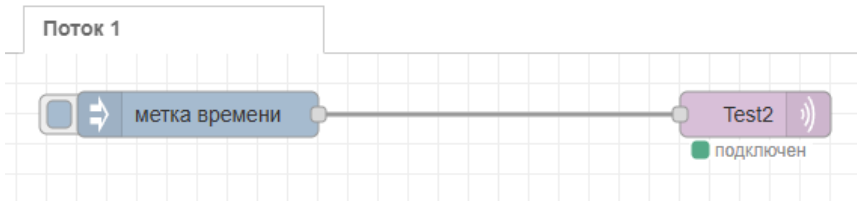


Рис. 3.6. Публикация сообщений в MQTT брокер

2. Для узла MQTT Out необходимо настроить соединение с брокером MQTT, задать тему опубликованного сообщения, указать параметр QoS, установить значение хранить – «да», если нужно, чтобы последнее сообщение сохранялось в брокере (см. рис. 3.7).
3. Нажмите «Развернуть поток» (Deploy).

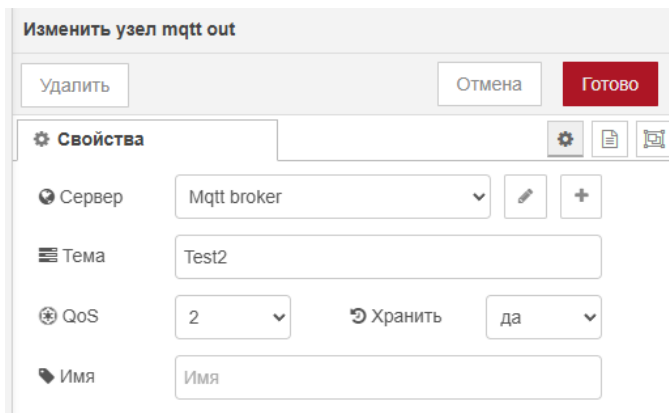


Рис. 3.7. Окно настроек узла MQTT Out

3.3. Промышленные протоколы связи

3.3.1. Интеграция с Modbus (RTU / TCP)

Шлюз выступает в роли Modbus Master (ведущего устройства), иницилирующего опрос периферийных модулей ввода и вывода (Modbus Slaves). Вся логика опроса, первичной обработки данных и маршрутизации реализована во встроенной среде визуального программирования Node-RED на базе палитры node-red-contrib-modbus.

3.3.1.1 Modbus RTU

Modbus RTU является одним из наиболее широко используемых протоколов связи в промышленной автоматизации. Он позволяет считывать данные с датчиков, контролировать состояние оборудования и управлять устройствами с помощью простой архитектуры «ведущий-ведомый».

Чтение данных с устройства

Для чтения данных с устройства выполняйте следующие:

1. Добавьте узел Modbus – Read (чтение данных с ведомого устройства) на рабочую область. Подключите его к узлу debug (см. рис. 3.8).

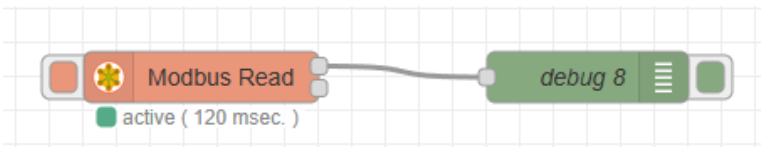


Рис. 3.8. Чтение данных с устройства по Modbus RTU

2. Дважды кликните мышкой по узлу Modbus – Read, чтобы открыть окно настроек (см. рис. 3.9). Введите Unit ID, выберите тип данных, которые вы хотите прочитать, укажите начальный адрес, установите количество значений для чтения и определите частоту опроса (как часто данные должны быть прочитаны).

Рис. 3.9. Окно настроек узла Modbus RTU – Read

3. Тип данных может принимать следующие значения:
 - **FC 1:** Read Coil Status – чтение DO.
 - **FC 2:** Read Input Status – чтение DI.
 - **FC 3:** Read Holding Registers – чтение АО.
 - **FC 4:** Read Input Registers – чтение AI.
4. Нажмите на значок + рядом с полем «Сервер», чтобы добавить сведения о подключении к Modbus (см. рис. 3.10);
5. В окне настройки сервера: задайте значение для параметра Type.
 - В поле Serial port выберите порт, к которому подключено устройство Modbus в формате /dev/com1;

3. Основной функционал работы Node RED

- Настройте параметры связи в соответствии с вашим устройством. Эти значения должны точно совпадать с конфигурацией устройства Modbus, в противном случае связь не будет установлена.

Изменить узел Modbus-Read > Изменить узел modbus-client

Удалить Отмена Обновить

Свойства

Settings Queues Optionals

Имя Modbus RTU

Type Serial Expert

Serial port /dev/com1

Serial type RTU

Baud rate 57600

Data Bits 8

Stop Bits 1

Parity None

Connection delay (ms) 100

Unit-Id 2

Timeout (ms) 1000

Reconnect on timeout ☒

Reconnect timeout (ms) 200

Рис. 3.10. Настройки сервера Modbus RTU – Read

6. Нажмите «Развернуть поток» (Deploy). Если все настроено правильно, вы начнете видеть данные с устройства Modbus в режиме реального времени на боковой панели отладки.

Запись данных в устройство

Modbus RTU позволяет записывать данные обратно на устройства.

1. Добавьте узел Modbus Write на рабочую область. Этот узел будет отправлять данные на ваше устройство Modbus.
2. В окне настроек введите: Unit ID устройства, выберите тип данных, которые вы хотите записать, укажите начальный адрес (см. рис. 3.11).
3. Нажмите на значок + рядом с полем «Сервер», чтобы добавить сведения о подключении к Modbus RTU.

Рис. 3.11. Окно настроек узла Modbus RTU – Write

4. Подключите узел Inject к узлу Modbus Write для отправки значений. Нажмите «Развернуть поток» (Deploy).

3.3.1.2 Modbus TCP

Чтение данных с устройства

1. Добавьте узел Modbus – Read на рабочую область. Подключите его к узлу debug (см. рис. 3.12).

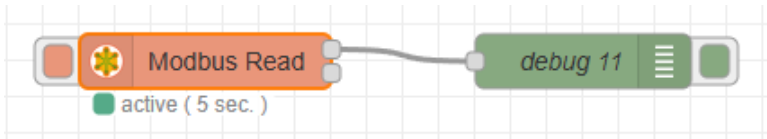


Рис. 3.12. Чтение данных с устройства по Modbus TCP

2. Дважды кликните мышкой по узлу Modbus – Read, чтобы открыть окно настроек (см. рис. 3.13).
3. Введите Unit ID, выберите тип данных, которые вы хотите прочитать, укажите начальный адрес, установите количество значений для чтения и определите частоту опроса (как часто данные должны быть прочитаны).

Рис. 3.13. Окно настроек узла Modbus TCP – Read

4. Нажмите на значок + рядом с полем «Сервер», чтобы добавить сведения о подключении к Modbus TCP.
5. В окне настройки сервера задайте: IP-адрес и номер порта (порт 502 - является портом по умолчанию для Modbus), тип TCP и идентификатор устройства (см. рис. 3.14).

Рис. 3.14. Настройки сервера Modbus TCP – Read

6. Нажмите «Развернуть поток» (Deploy). Вы начнете видеть данные с устройства Modbus в режиме реального времени на боковой панели отладки (см. рис. 3.15).

```
20.11.2025, 16:41:12 node: debug 8
Test3 : msg.payload : array[8]
  ► [ false, true, false, false, false,
    true, false, false ]
```

Рис. 3.15. Вывод данных, считанных с устройства

Запись данных в устройство

1. Добавьте узел Modbus Write на рабочую область. Этот узел будет отправлять данные на ваше устройство Modbus.

3. Основной функционал работы Node RED

2. Подключите узлы Inject к узлу Modbus Write для отправки значений true и false (см. рис. 3.16).

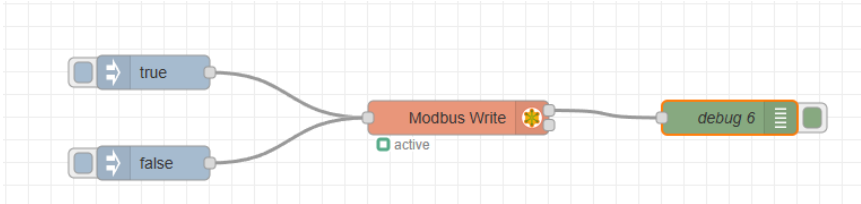


Рис. 3.16. Запись данных на устройство по Modbus TCP

3. Дважды кликните мышкой по узлу Modbus Write, чтобы открыть окно настроек (см. рис. 3.17).
4. Введите Unit ID, выберите тип данных, которые вы хотите записать, укажите начальный адрес.
5. Тип данных может принимать следующие значения:
 - **FC 5:** Force Single Coil – запись одного DO.
 - **FC 6:** Preset Single Register – запись одного АО.
 - **FC 15:** Force Multiple Coils – запись нескольких DO.
 - **FC 16:** Force Multiple Registers – запись нескольких АО.

The screenshot shows the 'Изменить узел Modbus-Write' (Edit Modbus-Write Node) configuration window. At the top, there are buttons for 'Удалить' (Delete), 'Отмена' (Cancel), and 'Готово' (Done). Below these is a 'Свойства' (Properties) tab. Under the 'Settings' sub-tab, there are several input fields: 'Name' (with a placeholder 'Name'), 'Unit-Id' (with the value '1'), 'FC' (a dropdown menu currently showing 'FC 5: Force Single Coil'), and 'Address' (with the value '5'). There is also a checkbox for 'Delay to activate input' which is currently unchecked. At the bottom, there is a 'Server' dropdown menu showing 'Modbus TCP Write', followed by edit and add icons.

Рис. 3.17. Окно настроек узла Modbus TCP – Write

3. Основной функционал работы Node RED

- Нажмите на значок + рядом с полем «Сервер», чтобы добавить сведения о подключении к Modbus TCP.
- Настройте узлы Inject для отправки значений true и false (см. рис. 3.18).

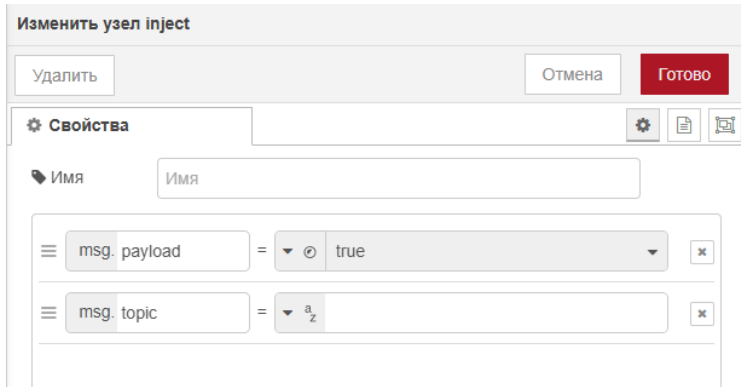


Рис. 3.18. Окно настроек узла Inject

- Нажмите «Развернуть поток» (Deploy). Произойдет запись данных на устройство Modbus (см. рис. 3.19).

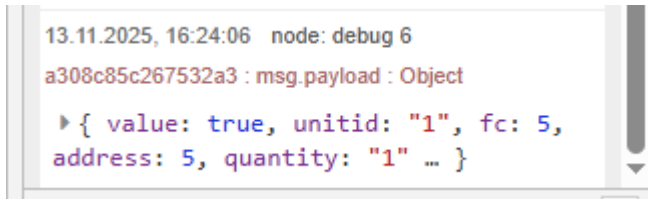


Рис. 3.19. Запись данных на устройство Modbus

3.3.2. Работа шлюза как OPC UA-сервер

Рассмотрим пример настройки OPC UA сервера (OpcUA-compact-server) с использованием Node-RED.

Добавьте узлы Inject, Function и Debug на рабочую область, для генерации 8 случайно сгенерированных Inputs и Outputs значений и сохранения их в виде потоковых переменных (см. рис. 3.20).

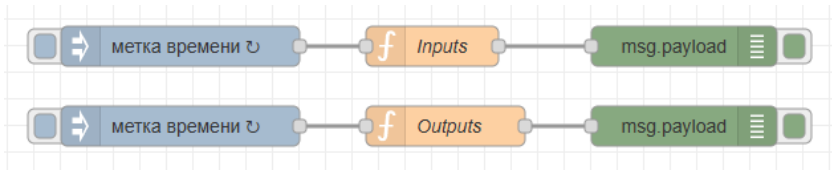


Рис. 3.20. Вывод информации о значениях тегов

В Node-RED функция `flow.set()` используется для хранения данных в контексте потока, позволяя сохранять информацию, доступную любому узлу внутри того же потока. Это особенно полезно, когда нужно обмениваться данными между разными частями вашего потока.

Значения потоковых переменных хранятся в памяти, щелкнув значок сворачиваемой стрелки на правой боковой панели и выбрав «Контекстные данные» (см. рис. 3.21) открывается список потоковых значений (см. рис. 3.22).

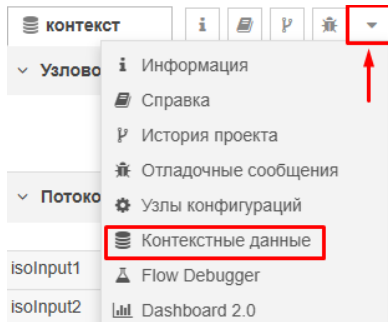


Рис. 3.21. Открытие контекстных данных

При нажатии на кнопку «обновить», список отображаемых Inputs и Outputs значений потоковых переменных меняется (см. рис. 3.22).



контекст

Узловой

ничего не выбрано

Потоковый

11.12.2025, 09:34:59

isoInput1	50.440475721280244
isoInput2	12.074009233835874
isoInput3	13.405493730445745
isoInput4	14.404319295072877
isoInput5	15.322563209972197
isoInput6	16.931189281604997
isoInput7	17.83208387063274
isoInput8	18.417939939359883
isoOutput1	10.901743279503556
isoOutput2	2.3395770108067087
isoOutput3	3.8334593785208533
isoOutput4	4.196468436145752
isoOutput5	5.286898098136687
isoOutput6	6.0667363899637605
isoOutput7	7.150803029065509
isoOutput8	8.326094674390896

Глобальный

Рис. 3.22. Вывод информации о потоковых значениях

3.3.1.3 Настройка OPC-сервера

1. Для этого потребуется установить необходимый пакет узлов: **node-red-contrib-opcua-server**
2. Перетащите узел OpcUA-compact-server на рабочую область (см. рис. 3.23 **Ошибка! Источник ссылки не найден.**).

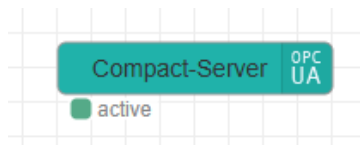


Рис. 3.23. Настройка узла OpcUA-compact-server

3. Дважды кликните по узлу OpcUA-compact-server, чтобы открыть настройки конфигурации. В панели конфигурации (Settings) введите номер порта (см. рис. 3.24).

3. Основной функционал работы Node RED

The screenshot shows the 'Изменить узел орсua-compact-server' (Edit Compact Server) dialog box in Node-RED. The 'Свойства' (Properties) tab is active, and the 'Settings' sub-tab is selected. The 'Port' field is highlighted with a red box and contains the value '54845'. Other fields include 'Resource Path' (UA/NodeRED/Compact), 'Product URI' (NodeOPCUA-Server-(port)), 'Alternate Hostname', 'Init Timeout' (1000), 'Shutdown Timeout' (100), 'Close Timeout' (200), 'Имя' (Name), 'Auditing' (unchecked), 'Show Activities' (unchecked), and 'Show Errors' (checked, highlighted with a red box).

Рис. 3.24. Ввод номера порта на вкладке Settings

4. Переключитесь на вкладку «Discovery» в настройках конфигурации. Здесь вам нужно создать URL Endpoint для сервера OPC-UA (см. рис. 3.25).

The screenshot shows the same 'Изменить узел орсua-compact-server' dialog box, but with the 'Discovery' sub-tab selected. The 'Register Server Method' is set to 'HIDDEN'. A list explains the methods: HIDDEN (server doesn't expose itself), MDNS (server publishes to mDNS), and LDS (server registers to LDS or LDS-ME). The 'Discovery' checkbox is checked. The 'Endpoint Uri' field is highlighted with a red box and contains the value 'opc.tcp://192.168.10.122:54845'. The 'MDNS capabilities' field contains 'NA,DA,...'.

Рис. 3.25. Ввод информации о URL

5. На вкладке «Security» по умолчанию активен анонимный доступ (для производственной системы позже можно установить настройки безопасности) (см. рис. 3.26).

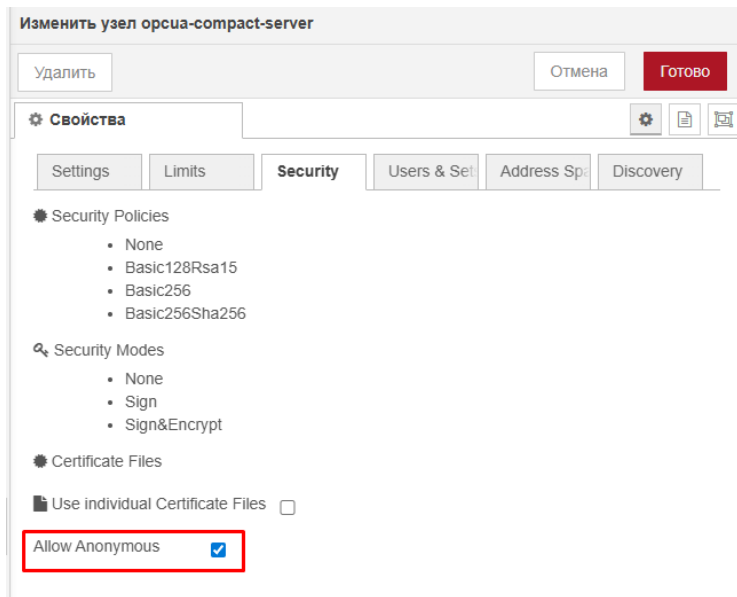


Рис. 3.26. Настройки безопасности

Вкладка «Адресное пространство» – это место, где строится информационная модель сервера OPC UA. Эта модель определяет структуру и данные, доступные клиентам OPC UA (см. рис. 3.27).

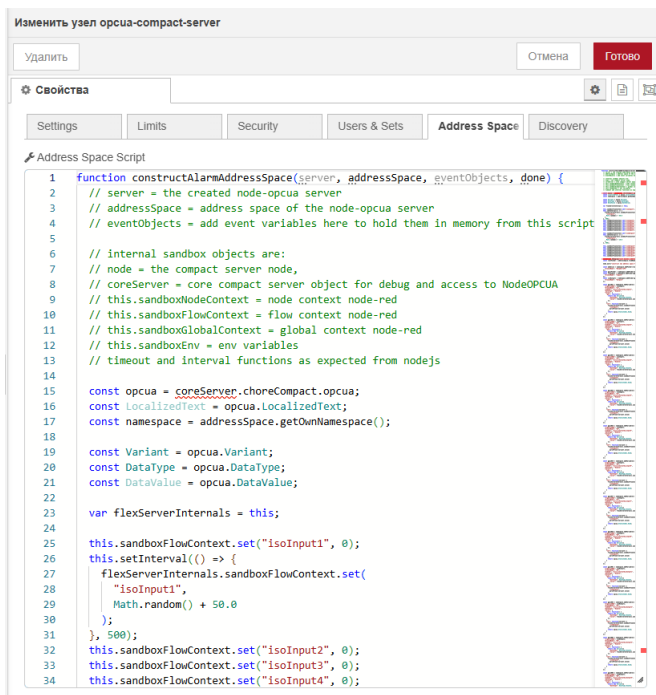


Рис. 3.27. Настройка адресного пространства

3.3.3. Работа шлюза как OPC UA-клиент

Рассмотрим интеграцию OPC UA с использованием Node-RED. Это позволит считывать значения в реальном времени с ПЛК, датчиков и промышленных устройств.

Чтобы работать с OPC UA в Node-RED, сначала нужно установить необходимые узлы. Для установки пакета узлов необходимо в меню в правом верхнем углу выбрать «Управление палитрой», перейти на вкладку «Установить» и выполнить поиск по запросу:

node-red-contrib-opcua

В качестве OPC UA сервера будем использовать MasterOPC Universal Modbus Server (см. рис. 3.28).

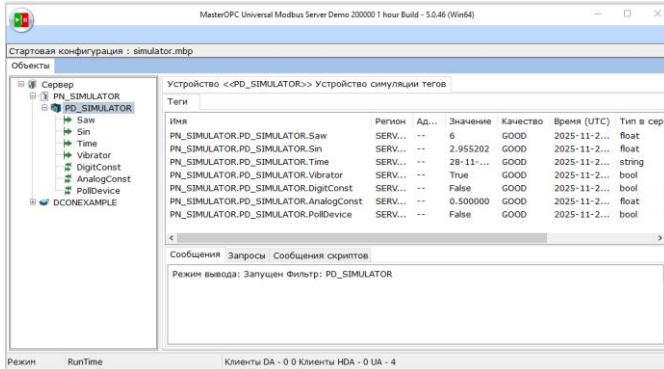


Рис. 3.28. MasterOPC Universal Modbus Server

3.3.1.4 Информации о структуре тегов

Если вы ещё не знаете идентификаторы узлов тегов, к которым хотите получить доступ, используйте узел OPC UA Browser для изучения структуры тегов.

1. Перетащите узлы: Inject, OPC UA Browser и Debug.
2. Подключите выход узла Inject ко входу узла браузера (OPC UA Browser), затем подключите выход браузера к узлу отладки (см. рис. 3.29).

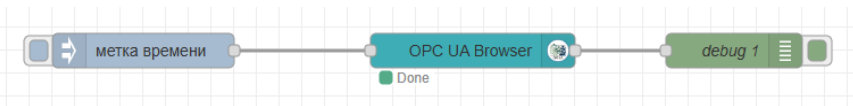


Рис. 3.29. Изучение структуры тегов при помощи узла OPC UA Browser

3. Дважды кликните мышкой по узлу OPC UA Browser, чтобы открыть окно настроек (см. рис. 3.30).

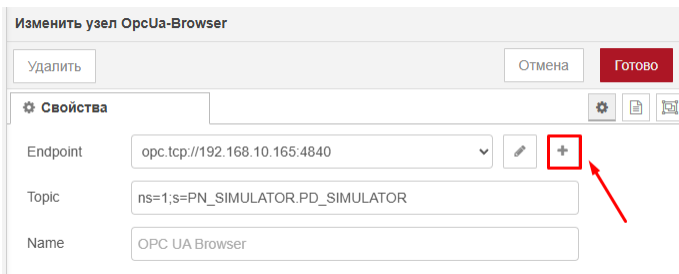


Рис. 3.30. Окно настроек узла OPC UA Browser

3. Основной функционал работы Node RED

4. В узле Браузер нажмите на значок +, чтобы создать новую конфигурацию конечной точки. Введите адрес вашего сервера OPC UA, например: `opc.tcp://192.168.10.165:4840`.
5. Установите режим безопасности на None (можно добавить защиту позже) (см. рис. 3.31).

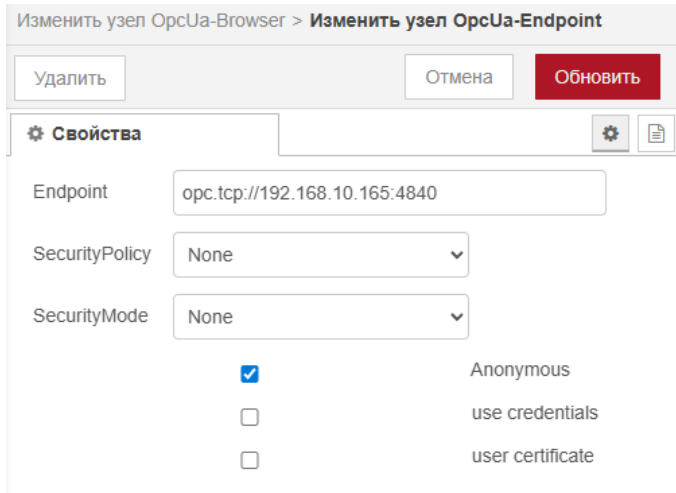


Рис. 3.31. Настройка конфигурации конечной точки

6. В узле Браузер установите тему на (корневую папку Objects) в формате: `ns=1;s=PN_SIMULATOR.PD_SIMULATOR`.
7. Настройте узел Inject для отправки временной метки. Нажмите «Развернуть поток» (Deploy) и кликните по узлу Inject.
8. На боковой панели отладки отобразится информация о структуре тегов (см. рис. 3.32). Теперь вы можете определить точные идентификаторы узлов для чтения или записи.

```
28.11.2025, 15:36:11 node: debug 1
msg.payload : array[7]
  ▼ array[7]
    ▼ 0: object
      ▼ item: object
        referenceTypeId: "ns=0;i=46"
        isForward: true
        nodeId:
          "ns=1;s=PN_SIMULATOR.PD_SIMULATOR.Saw"
        ▶ browseName: object
        ▶ displayName: object
        nodeClass: 2
        typeDefinition: "ns=0;i=63"
        value: 6
        dataType: "Float"
      ▶ 1: object
      ▶ 2: object
      ▶ 3: object
      ▶ 4: object
      ▶ 5: object
      ▶ 6: object
```

Рис. 3.32. Вывод данных о структуре тегов

Когда вы узнаете идентификаторы узлов, можно начинать читать данные с промышленного оборудования через сервер OPC UA.

3.3.1.5 Чтение одного тега

1. Перетащите узлы: Inject, OPC UA-Item, OPC UA Client и Debug на рабочую область (см. рис. 3.33).



Рис. 3.33. Чтение одного тега

2. В окне настроек узла OPC UA-Item задайте: идентификатор узла (Item) в формате ns=1;s=PN_SIMULATOR.PD_SIMULATOR.Saw и тип данных (см. рис. 3.34). Информация об идентификаторе узла отображается в поле nodeId (см. рис. 3.35).

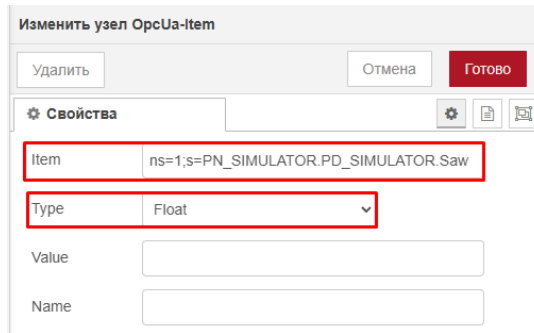


Рис. 3.34. Окно настроек узла OPC UA-Item

3. В окне настроек узла OPC UA Client задайте адрес вашего сервера OPC UA и установите действие на READ (см. рис. 3.35).

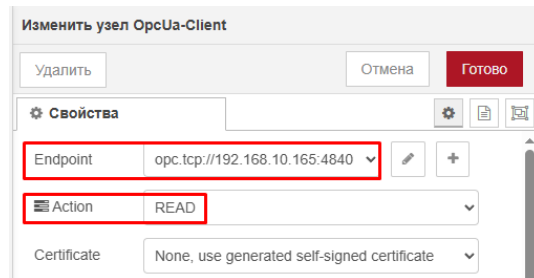


Рис. 3.35. Окно настроек узла OPC UA Client

4. Нажмите «Развернуть поток» (Deploy) и кликните по узлу Inject для запуска операции чтения тега. На боковой панели отладки отобразится значение тега (см. рис. 3.36).

```
28.11.2025, 16:21:34 node: Tag Value
ns=1;s=PN_SIMULATOR.PD_SIMULATOR.Saw :
msg.payload : number
5
```

Рис. 3.36. Вывод информации о значении тега

3.3.1.6 Чтение нескольких тегов

1. Для реализации пакетного считывания данных (см. рис. 3.37) в окне настроек узла OPC UA Client установите действие на READ MULTIPLE (см. рис. 3.38).

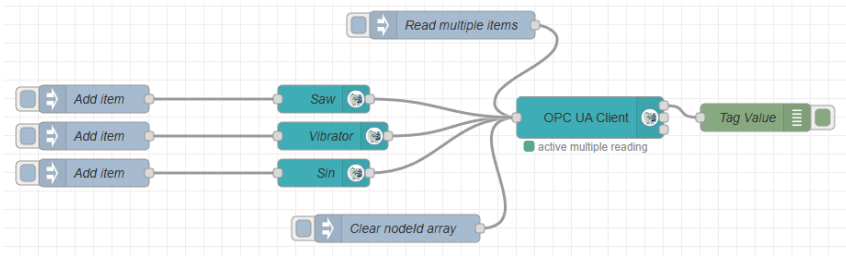


Рис. 3.37. Чтение нескольких тега



Рис. 3.38. Окно настроек узла OPC UA Client

2. Добавьте узлы Inject и OPC UA Item для каждого тега, который хотите прочитать. Настройте узлы Inject для отправки значений типа строка (см. рис. 3.39).

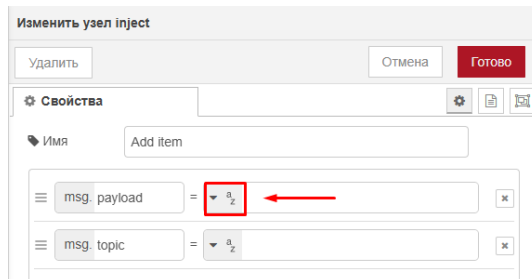


Рис. 3.39. Окно настроек узла Inject

3. Основной функционал работы Node RED

3. На вход узла OPC UA Client добавьте узлы Inject для чтения нескольких тегов (см. рис. 3.40) и очистки массива идентификаторов узлов (см. рис. 3.41).

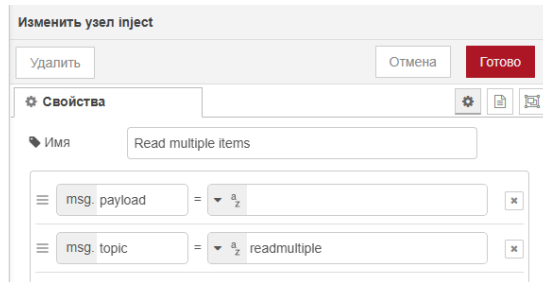


Рис. 3.40. Окно настроек узла Inject для чтения нескольких тегов

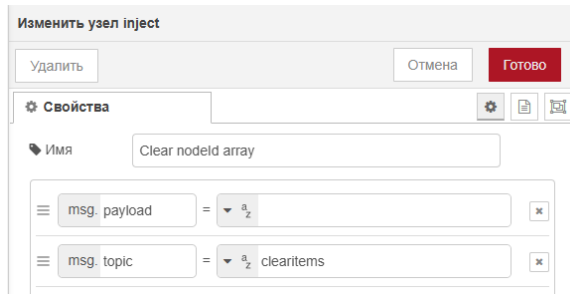


Рис. 3.41. Окно настроек узла Inject очистки массива идентификаторов узлов

4. Нажмите «Развернуть поток» (Deploy) и кликните по каждому узлу Inject. Кликните по узлу чтения нескольких тегов для запуска операции чтения тегов. На боковой панели отладки отобразятся значения тегов (см. рис. 3.42).

```
28.11.2025, 17:08:58 node: Tag Value
[object Object] : msg.payload : number
10

28.11.2025, 17:08:58 node: Tag Value
[object Object] : msg.payload : boolean
true

28.11.2025, 17:08:58 node: Tag Value
[object Object] : msg.payload : number
-8.83454704284668
```

Рис. 3.42. Вывод информации о значениях тегов

3.3.4. Настройка отправки уведомлений на Email

В Node-RED можно настроить автоматическую от отправку уведомлений на вашу электронную почту, например, если показания датчика выше или ниже определённого порога, или настроить от отправку уведомлений по расписанию.

Установка и настройка ноды email

Для этого необходимо установить необходимый пакет узлов: **node-red-node-email**.

Добавьте узел email на рабочую область. Подключите его к узлу inject (см. рис. 3.43).

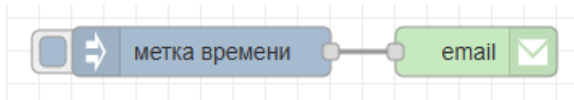


Рис. 3.43. Настройка уведомлений на почту

Для настройки email-оповещений в Node-RED потребуется:

- **Аккаунт отправителя:** рекомендуется создать отдельный почтовый ящик, предназначенный для системных уведомлений.
- **Пароль приложения:** специальный пароль, который генерируется в учётной записи отправителя вместо основного пароля (необходимо для авторизации Node-RED).

ВАЖНО! Для создания пароля приложения в учётной записи должна быть включена двухфакторная аутентификация (2FA).

Создание пароля приложения

Для создания пароля приложения (на примере почты Mail) необходимо выполнить следующую последовательность действий:

1. Откройте почту.
2. В левом нижнем углу перейдите в Настройки → Все настройки → Безопасность → Пароли для внешних приложений;
3. Нажмите «Создать»;
4. Укажите название приложения, для которого создаете пароль (например, Node-RED). С этим названием пароль будет отображаться в списке;
5. Нажмите «Продолжить»;

3. Основной функционал работы Node RED

6. Выберите тип протокола. От типа протокола зависит, к каким вашим данным в Mail почтовая программа получит доступ. Выбирайте протокол, который даёт доступ только к нужным данным — это повысит безопасность почты;
7. Нажмите «Продолжить»;
8. Пароль приложения отобразится в новом окне, скопируйте и сохраните его. Таким образом, появится пароль приложения для использования в Node-RED для отправки уведомлений на электронную почту.

ВАЖНО! Созданный пароль можно увидеть только один раз. Если вы ввели его неправильно и закрыли окно, удалите текущий пароль и создайте новый.

ВАЖНО! Если вы используете другого поставщика электронной почты, проверьте, что нужно сделать для создания пароля приложения.

9. В окне настроек узла email введите электронную почту получателя (✉ To), данные о SMTP Server, SMTP port, электронную почту отправителя (Userid) соответствующий пароль приложения (SMTP Пароль приложения), который вы сгенерировали ранее (Password) (см. рис. 3.44).

Изменить узел email

Удалить Отмена Готово

Свойства

Имя

To email@address.com

Server smtp.mail.ru

Port 465 Use secure connection.

Auth type Basic

Userid r3surr3ct3d@mail.ru

Password

TLS option Check server certificate is valid

Рис. 3.44. Настройка уведомлений на почту

Если все указано корректно, то на электронную почту получателя придет текстовое уведомление.

4. 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.

4.1. 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).

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 (см. рис. 4.1):
 - **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. 4.1. 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.

Приём данных в Node-RED

Для приёма данных в среде Node-RED добавьте на рабочую область узел mqtt in, укажите параметры подключения к брокеру и целевой топик, после чего подключите узел function для разбора входящего пакета и извлечения значений сигналов. Более подробная информация о работе с протоколом MQTT и примеры создания потоков опроса приведены в подразделе 3.2.

4.2. 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.

Предварительное конфигурирование модулей

Перед организацией опроса в среде Node-RED 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 среде Node-RED строится на базе узлов палитры node-red-contrib-modbus. Подробное руководство по настройке узлов чтения и записи Modbus приведено в подразделе 3.3.1.

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

4.3. Модули с поддержкой LoRaWAN

Беспроводные модули RealLab с поддержкой LoRaWAN (в частности, термощтанга TR-120091-L) предназначены 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 среда Node-RED считывает информацию для обработки, логического анализа и отображения на мнемосхемах оператора.

Предварительное конфигурирование и регистрация датчиков в ChirpStack

Перед организацией сбора данных в среде Node-RED необходимо зарегистрировать беспроводное устройство и настроить декодер полезной нагрузки в локальном сетевом сервере ChirpStack:

1. Подключение к сетевому серверу:
 - Откройте веб-браузер и перейдите в веб-интерфейс сетевого сервера LoRaWAN по адресу: `http://<IP-адрес_шлюза>:8081` (или через системный конфигуратор по порту 8080).
 - Авторизуйтесь в системе (по умолчанию: имя пользователя - admin, пароль - admin).
2. Настройка и регистрация термощтанга в сетевом сервере ChirpStack.
 - Создание профиля и настройка встроенного декодера (Payload Codec).
 - Регистрация и настройка декодера термощтанга в ChirpStack.

Примечание: Подробное описание процедуры добавления устройств различных типов, структуры радиопакетов и настройки сетевых ключей приведено в [руководстве пользователя по работе с ПО шлюзов RealLab!](#) серии NLS-Gateway в разделах 3.5 и 3.6.

Организация приёма данных и визуализации в среде Node-RED

Для построения системы оперативного мониторинга параметров термостанги в среде Node-RED используются узлы подписки MQTT и библиотека виджетов node-red-dashboard:

1. Добавьте на рабочую область узел mqtt in.
2. В параметрах узла укажите локальный брокер шлюза (127.0.0.1:1883) и топик получения телеметрии.
3. Подключите узел function для распределения принятых параметров по графическим виджетам интерфейса.
4. Нажмите кнопку «Развернуть» (Deploy). Для перехода в графический интерфейс оператора откройте в браузере страницу: **Ошибка! Недопустимый объект гиперссылки..**

Пример потока для мониторинга параметров датчиков термостанги TR-120091-L (см. рис. 4.2).

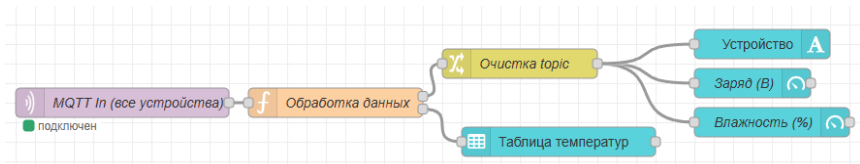


Рис. 4.2. Мониторинг параметров термостанги

Окно визуализации «Мониторинг датчиков термостанги» отображает следующие параметры: наименование устройства, время получения данных, данные с датчиков температуры, значение заряда и влажности (см. рис. 4.3).

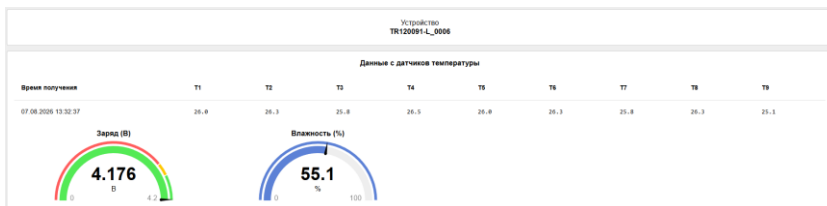


Рис. 4.3. Окно визуализации параметров термостанги

Примечание: Готовый пример проект(flow.json) с настроенным интерфейсом Dashboard и скриптами декодирования для термостанги TR-120091-L

доступен для загрузки на странице шлюза на официальном сайте компании [RealLab!](#).

5. 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.

5.1. 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 Node-RED).

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.

5.1.1. Просмотр информации о системе

Подключаете к веб-серверу RealLab конфигуратора для доступки, 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 (см. рис. 5.1):

- 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).

5. Диагностика и устранение неисправностей

Эти показатели позволяют быстро оценить, не перегружено ли шлюз, и принять решение о перезапуске служб или оптимизации проектов (например, в Node-RED или CODESYS).

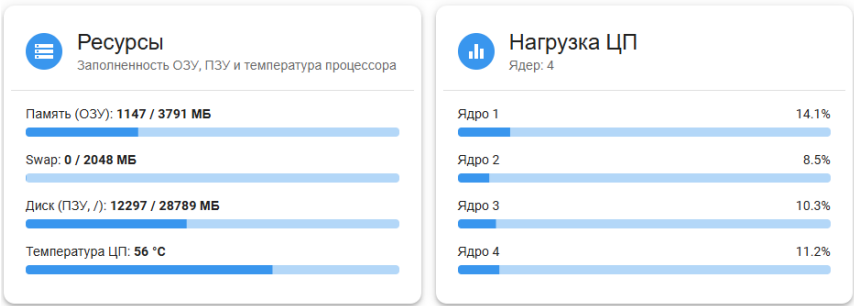


Рис. 5.1. Информация о нагрузке на шлюз

5.1.2. Управление процессами

Страница «Процессы» выводит список всех запущенных процессов (см. рис. 5.2) с их идентификаторами (PID), потреблением CPU и памяти. Процессы отсортированы по убыванию нагрузки на процессор, что облегчает поиск «тяжёлых» задач.

The screenshot shows the 'Диспетчер процессов' (Process Manager) interface. It includes a table of running processes and control buttons. The table has columns for PID, Name, Status, CPU %, MEM %, MEM (MB), and Command. The processes listed are codesyscontrol.bin, RLWebConfig, python3, chirpstack-concentratord-sx1302, and codesysedge.bin.

PID	Имя	Статус	CPU %	MEM %	MEM (MB)	Команда
35978	codesyscontrol.bin	sleeping	12.4	2.3	88.2	/opt/codesys/bin/codesyscontrol.bin /etc/codesyscontrol/CODESYSControl.cfg
1119	RLWebConfig	running	3.6	1.7	63.3	/usr/bin/rfwebconfig/RLWebConfig
1560	python3	sleeping	2.7	0.3	11.7	/home/pi/.venv/bin/python3 /usr/local/bin/gpio2btn.py
884	chirpstack-concentratord-sx1302	sleeping	2.7	0.1	3	/usr/bin/chirpstack-concentratord-sx1302 -c /etc/chirpstack-concentratord/region_ru864.toml -c /etc/chirpstack-concentratord/channels_ru864.toml
1071	codesysedge.bin	sleeping	2.3	0.3	9.5	/opt/codesysedge/bin/codesysedge.bin /etc/codesysedge/Gateway.cfg

Рис. 5.2. Диспетчер процессов в веб-конфигураторе на шлюзе

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 Node-RED или интенсивным сетевым обменом.

5.2. Просмотр системных журналов и диагностика ошибок

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

5.2.1. 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;
- **Node-RED** – журнал службы nodered.service;
- **CODESYS** – журнал службы CODESYS.

5.2.2. Рекомендации по анализу логов

При поиске неисправностей обращайтесь внимание на следующее:

5. Диагностика и устранение неисправностей

- **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!.

5.3. Типовые неисправности и методы их устранения

В табл. 2 приведены наиболее часто встречающиеся проблемы, их вероятные причины и рекомендуемые действия по восстановлению работоспособности.

Табл. 2. 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 Node-RED не запускается или работает нестабильно	Ошибка в потоках (flow) или в установленных узлах; недостаточно оперативной памяти; конфликт портов	Просмотреть логи Node-RED через веб-конфигуратор. При необходимости отключить автозагрузку, очистить проект (с созданием резервной копии) и попытаться запустить заново. Если проблема повторяется – оптимизировать потоки.

5. Диагностика и устранение неисправностей

Неисправность	Возможная причина	Действия по восстановлению
Нет связи с LoRaWAN-датчиками	Не совпадает Gateway ID в ChirpStack; неправильный частотный план; проблемы с радио-модулем	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.
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