

RS-485

Contents

Описание

Как правильно проложить шину

Добавление устройства в веб-интерфейс

Как ускорить опрос устройств

Работа с портом RS-485 контроллера из собственного ПО

Описание

RS-485 — стандарт коммуникации по двухпроводной шине.

Теоретически на шину можно подключать до 256 устройств. Длина линии может быть до 1200 метров, но она сильно влияет на скорость передачи данных.

Энциклопедия АСУ ТП. Интерфейс RS-485 (https://www.bookasutp.ru/Chapter2_3.aspx) — подробно про работу интерфейса.

В устройствах Wiren Board используется Протокол Modbus поверх RS-485. Пожалуйста, ознакомьтесь с ним для лучшего понимания работы устройств.

Максимальная скорость передачи данных в периферийных устройствах Wiren Board — до 115 200 бит/с.

Как правильно проложить шину

В статье RS-485:Физическое подключение описано как правильно проложить шину.

Добавление устройства в веб-интерфейс

RS-485:Настройка через веб-интерфейс — что сделать для появления устройства в веб-интерфейсе контроллера.

Как ускорить опрос устройств

Для ускорения опроса устройств по шине RS-485 рекомендуем:

1. Увеличить скорость обмена до 115200 бит/с. На разумных длинах и топологии сети все должно нормально работать. Если на шине есть устройства, не поддерживающие эту скорость, см. пункт 3.
2. Отключить через веб-интерфейс в настройках устройства ненужные каналы.
3. Разделить устройства по типам и портам, контроллере 2 порта RS-485 и еще 3 можно добавить модулями расширения:
 - Устройства, не поддерживающие скорость 115200, подключите отдельно.
 - Счетчики MAP так же подключите отдельно или с оборудованием, не требующим быстрой реакции. В счетчиках очень много параметров, опрос идет медленно.
 - При большом количестве устройств разделите их на несколько портов. При прочих равных скорость вырастет кратно количеству портов.

Работа с портом RS-485 контроллера из собственного ПО

- Стандартно в Wiren Board с подключёнными по RS-485 устройствами работает Драйвер `wb-mqtt-serial` (ранее `wb-homa-modbus`). Он позволяет работать с подключёнными устройствами RS-485 через систему MQTT-сообщений.
- Если вы хотите работать с портом RS-485 напрямую, не используя этот драйвер — отключите его, иначе он будет писать в порт RS-485.
- Работа с последовательным портом из Linux
- Доступ к порту RS-485 контроллера Wiren Board с компьютера
- Настройка параметров обмена данными по RS-485 для modbus-устройств Wiren Board

Веб-интерфейс Wiren Board

- English
- русский





Главная страница веб-интерфейса

Contents

Возможности

Как зайти в веб-интерфейс

Работа с веб-интерфейсом

Разделы интерфейса

Home (Главная страница)

Dashboards (Панели)

Devices (Устройства)

Widgets (Виджеты)

Пример создания виджетов

History (История показаний)

Rules (Правила-скрипты)

Settings -> Configs (Настройки -> Конфигурирование)

Settings -> WebUI (Настройки -> Веб-интерфейс)

Settings -> System (Настройки -> Системные)

Settings -> MQTT Channels (Настройки -> MQTT-каналы)

Settings -> Change access level (Настройки -> Права доступа)

Settings -> Logs (Настройки -> Логи)

Стандартные задачи, решаемые через веб-интерфейс

Подключить устройство RS-485 Modbus и создать кнопки управления на главной панели

Обновить прошивку контроллера

Облачный интерфейс

Настройка авторизованного доступа к веб-интерфейсу контроллера

Обновление веб-интерфейса

Основные отличия версии 2.x от 1.0

Возможности

Контроллер Wiren Board имеет встроенный веб-интерфейс. Через интерфейс можно:

- следить за состоянием контроллера и подключённых устройств и управлять ими;
- подключать устройства к контроллеру;
- настраивать контроллер и обновлять его ПО;
- писать правила на встроенном движке;
- настраивать SMS и email-уведомления;
- смотреть на графике историю значений (например, температуры).

Веб-интерфейс работает непосредственно на Wiren Board. В качестве веб-сервера работает nginx, сайт взаимодействует с MQTT через WebSocket.

Ниже дано описание версии 2.0. Про предыдущую версию можно найти на странице Веб-интерфейс Wiren Board 1.0.

Как зайти в веб-интерфейс

Чтобы зайти в веб-интерфейс контроллера Wiren Board, введите в адресную строку браузера IP-адрес контроллера.

Если вы находитесь в одной сети с контроллером и используете устройства Apple, компьютер с Linux или Windows 10 и выше — введите в адресную строку `wirenboard-XXXXXXX.local`, где XXXXXXXX — восьмизначный серийный номер контроллера.

Работа с веб-интерфейсом





Выбор уровня доступа

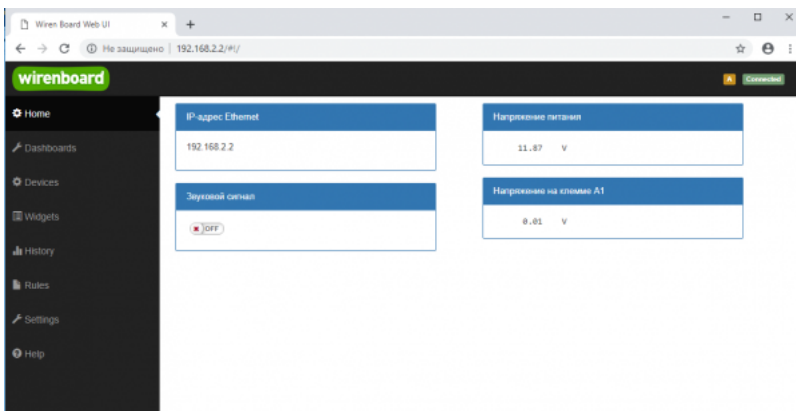
Для начала работы с веб-интерфейсом выберите уровень доступа. Для этого на вкладке **Settings** -> **Change access level** -> **Access Level** выберите один из пунктов **User**, **Operator** или **Administrator**. Уровни ограничивают доступ к функционалу веб-интерфейса: например, пользователь **User** может просматривать только настроенные виджеты, их редактирование и изменение настроек контроллера недоступно. Пользователь **Operator** получает доступ к контролам устройств, управляемых контроллером, может добавлять виджеты в панели (dashboards) (см. далее). Пользователь **Administrator** обладает всеми правами. Изменение текущего уровня доступа может быть изменено любым пользователем и предназначено больше для защиты от неверных действий, чем для разграничения прав.

В дальнейшем изложении мы предполагаем, что все действия выполняются пользователем **Administrator**.

Чтобы получить уровень доступа **Administrator**, на вкладке **Access Level** выберите опцию **Administrator**, подтвердите выбор, установив флажок в поле **"I take full responsibility for my actions"**, и нажмите кнопку **Apply** (как показано на рисунке "Выбор уровня доступа").

Разделы интерфейса

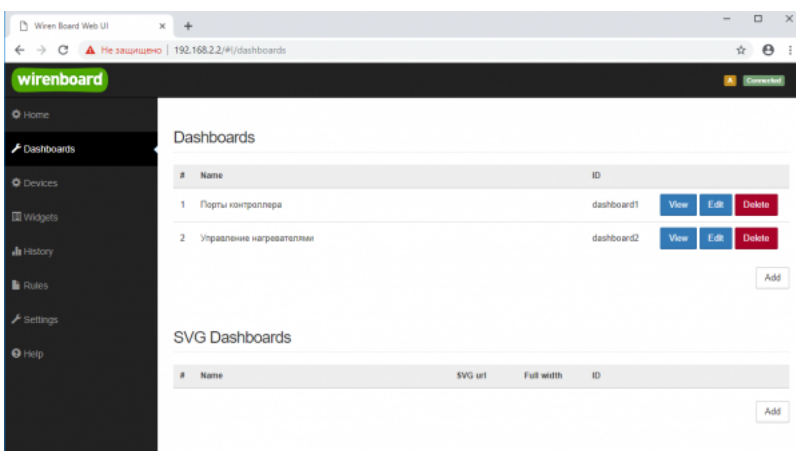
Home (Главная страница)



Home - главная страница

Это главная страница пользователя. На неё выводятся элементы интерфейса - так называемые "виджеты" (widget). Это могут быть показания датчиков (например, датчика температуры), кнопки включения света, управления подключёнными реле. Набор виджетов на главной странице полностью настраивается пользователем в меню **Settings** -> **Web UI** -> **Common Info**, где можно выбрать панель, которая будет отображаться во вкладке **Home** по умолчанию.

Dashboards (Панели)

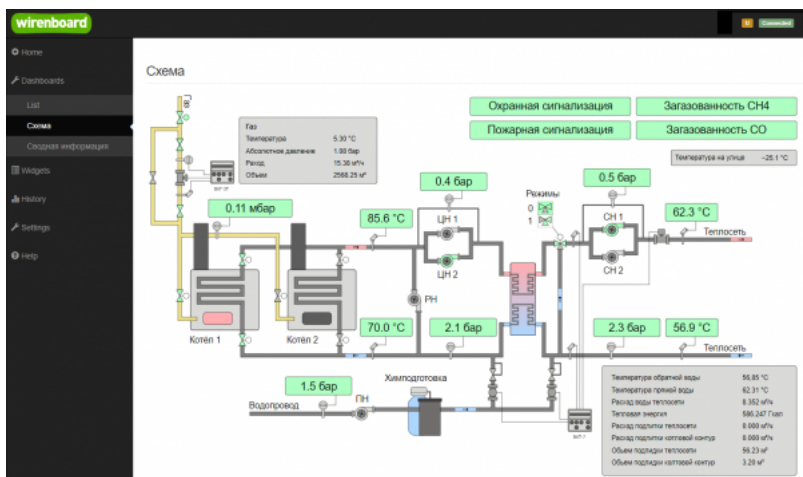


Dashboards - список панелей

Виджеты со схожим назначением можно группировать в панели, где на одном экране находятся все необходимые кнопки, настройки и показатели датчиков. Например, можно объединить виджеты включения подогрева, кондиционера, отображения температуры и влажности. В разделе **Dashboards** можно увидеть все созданные панели. Раздел Home тоже отображает одну из панелей, выбранную в настройках.

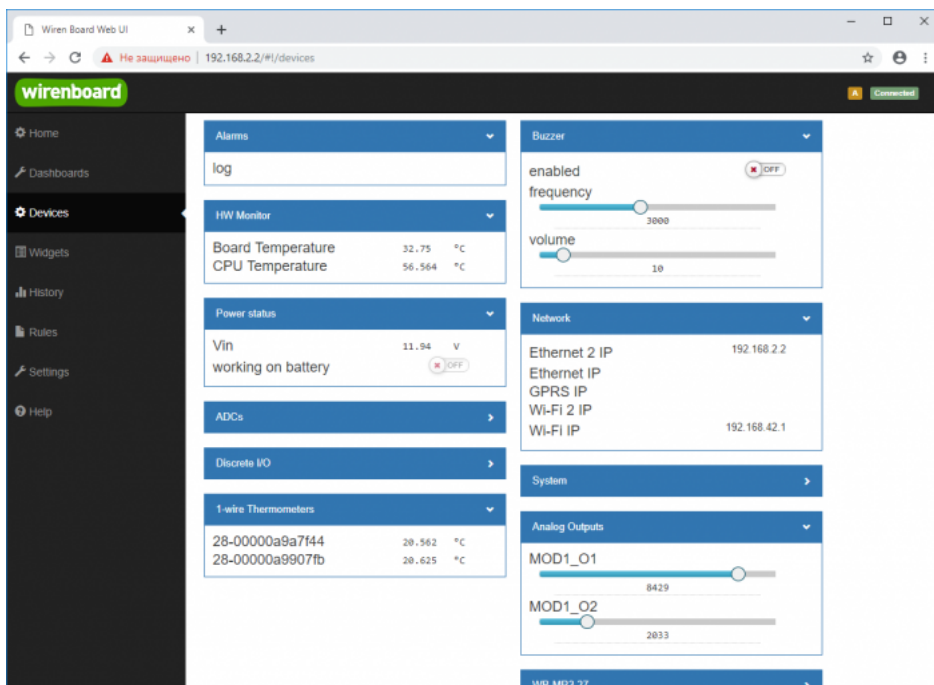
Дважды щелкнув по вкладке **Dashboards** на боковой панели, можно раскрыть список всех панелей, созданных в веб-интерфейсе.

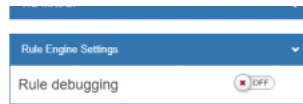
Кроме текстовых панелей с виджетами, можно создавать интерактивные SVG-панели (SVG Dashboards).



Пример SVG-панели

Devices (Устройства)





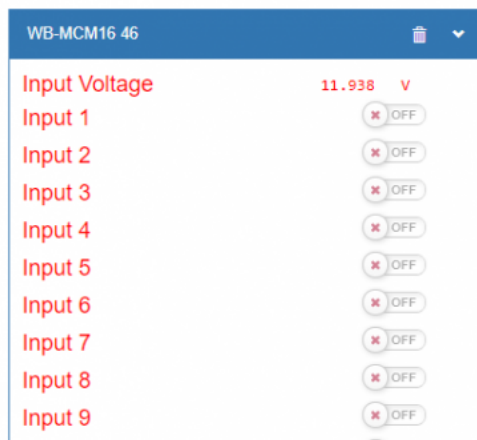
Devices - список всех аппаратных функций контроллера и подключённых устройств

На странице **Devices** отображаются все аппаратные возможности контроллера (состояние входов, выходов, напряжение питания), а также подключённых датчиков и устройств. Если вы подключили к контроллеру внешний модуль, все его меняющиеся значения будут отображены тут.

Каждый элемент устройства (показание значения напряжения, сетевой адрес, кнопка управления реле, флажок состояния входа и т.п.) -- называется "контрол". Несколько контролов могут быть объединены в один виджет. Подробнее смотрите в разделе Widgets (Виджеты).

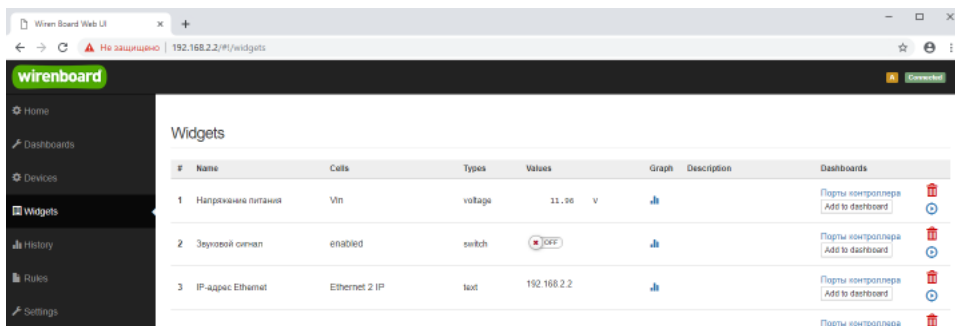
Подключаемые устройства (Modbus-модули, боковые и внутренние модули) **не** определяются контроллером автоматически. Чтобы на этой странице появились аппаратные возможности подключённых устройств (например, внешних модулей реле), сначала нужно настроить их через раздел Configs (Конфигурирование).

Удалить отключенные/неработающие устройства из веб-интерфейса можно с помощью кнопки **Delete** в виде значка с изображением мусорной корзины, в верхней строке плитки устройства. Кнопка появляется, когда указатель мыши находится над плиткой устройства.



Удаление отключенного устройства

Widgets (Виджеты)



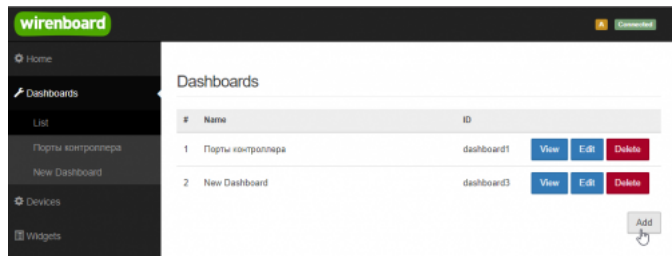
ID	Название	Тип	Значение	Статус	Действия
5	Температура ЦПУ / CPU Temperature	temperature	56.564 °C	↓	Add to dashboard
6	Комната 1	Температура воздуха	28.312 °C	↓	Управление нагревателем Add to dashboard
	Конвектор	switch	OFF	↓	
7	Комната 2	Температура воздуха	28.175 °C	↓	Управление нагревателем Add to dashboard
	Конвектор	switch	OFF	↓	

Widgets - страница управления виджетами

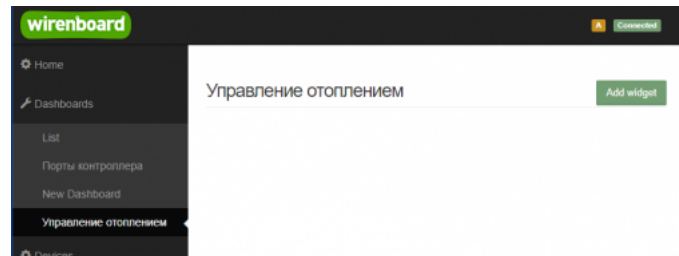
Виджеты - комбинированные элементы интерфейса контроллера, включающие в себя набор контролов, то есть аппаратных параметров контроллера и подключённых к нему устройств - тех, что отображаются на странице Devices (Устройства).

На странице Widgets представлен список всех виджетов, созданных в системе. Сами виджеты создаются в настройках панелей, на этой странице ими можно только управлять: просматривать, удалять и добавлять к существующим панелям-дашбордам.

Пример создания виджетов



Создать новую панель



Создать новый виджет

Для примера создадим два виджета с показаниями температуры и переключателями управления отопительными конвекторами для двух комнат "Комната 1" и "Комната 2".

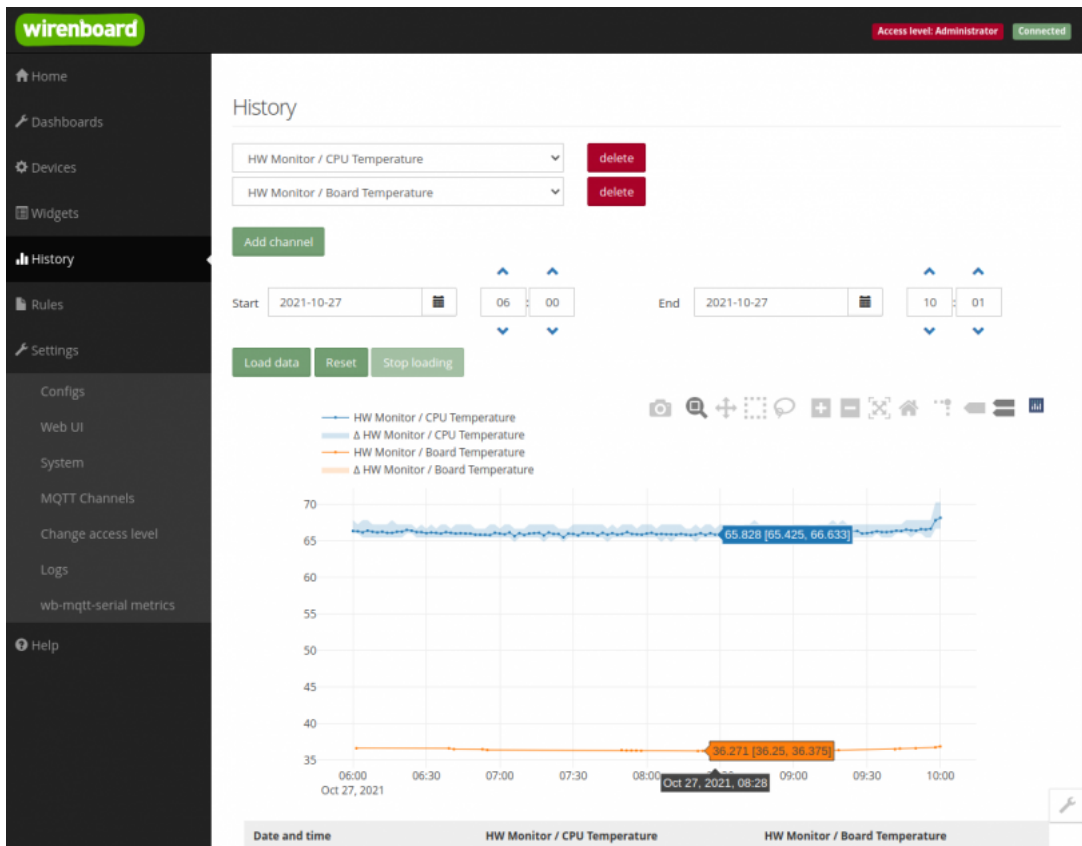
- На боковой панели щелкнем на вкладке **Dashboards**, в раскрывшемся списке выберем элемент **List** и на открывшейся странице нажмем кнопку **Add**.
- В поле **Name** напишем название новой панели, "Управление отоплением" и нажмем кнопку **Save**.
- В списке на странице **Dashboards** щелкнем по кнопке **View** напротив новой панели "Управление отоплением".
- В открывшемся окне с названием панели щелкнем по кнопке **Add widget** в правом верхнем углу окна (см. Рис. "Создать новый виджет").
- В заголовке виджета укажем название, в нашем случае "Комната 1", в списке **Add control...** выберем контрол, соответствующий термометру в первой комнате, еще раз в этом списке выберем реле, которое будет включать нагреватель.
- В поле **Name** виджета можно задать осмысленные названия для контролов, например: "Температура" и "Обогреватель". Снимите флажок **Compact mode**, чтобы эти названия контролов отображались в виджете.
- В поле **Widget description** можно написать назначение виджета.
- Аналогично создадим виджет для управления отоплением в комнате 2.
- Для внесения изменений подведите курсор к заголовку виджета и нажмите кнопку **Edit widget**, внесите изменения и нажмите кнопку **Save**.

Добавить новый контрол в виджет

Пример создания виджета

Пример готовых виджетов на панели

History (История показаний)



Пример отображения исторических данных

На странице *History* можно просмотреть историю изменения значений аппаратных ресурсов (например, датчиков температуры, напряжения, показаний счётчиков). История представляется одновременно в виде графика и таблицы значений с метками времени.

Возможности просмотра исторических данных:

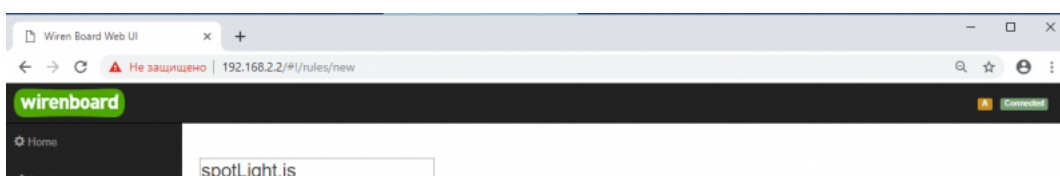
- Указание интервала времени для отображения данных
- Добавление и удаление нескольких показателей (кнопки Add channel и delete) на график
- Просмотр данных в виде графика и в виде таблицы
- Загрузка данных за выбранный период в csv-формате.

При наведении указателя мыши на область изображения становятся доступными дополнительные функции (кнопки в верхней правой части графика):

- Сохранение графика в формате .png
- Переключение между режимами Zoom (увеличения/уменьшения отрезка данных и масштаба с помощью выделения нужной области указателем мыши) и панорамирования Pan (перемещения области видимости с зажатой левой кнопкой мыши)
- Уменьшение и увеличение отображаемого временного интервала (Zoom in и Zoom out)
- Автоматический выбор масштаба графика по обеим осям
- Возвращение масштаба осей к исходному
- Включение/выключение указателя координат

Утилита для извлечения исторических данных из внутренней базы данных

Rules (Правила-скрипты)




```

Save
1 defineVirtualDevice("spotlight", {
2   title: "Spotlight", //
3   cells: {
4     //
5     on: {
6       type: "switch",
7       value: false,
8       readonly: true
9     }
10  };
11 });
12
13 defineRule("spotlightRelayChange", {
14   whenChanged: "mb-mr11_111/K5",
15   then: function (newValue, devName, cellName) {
16     // if (newValue) { dev["radioRemote"]["K4"]=dev["radioRemote"]["K4"];
17     log(devName);
18     dev["spotlight"]["on"]=newValue;
19   }
20 });
21
22 defineRule("spotlightRadioRemoteChange", {
23   whenChanged: "radioRemote/K1",
24   then: function (newValue, devName, cellName) {
25     // if (newValue) { dev["mb-mr11_111"]["K5"]=dev["mb-mr11_111"]["K5"];
26     log(devName);
27     dev["mb-mr11_111"]["K5"]=newValue;
28   }
29 });

```

Скрипт, открытый для просмотра и редактирования

На странице **Rules** можно создавать и редактировать правила. Правила пишутся на простом языке, похожем на JavaScript и позволяют создавать правила ("включай свет с 10:00 до 18:00") или виртуальные устройства (например, кнопка в интерфейсе, которая включает и отключает всё освещение в здании вместе).

- Подробнее про скрипты.

Settings -> Configs (Настройки -> Конфигурирование)

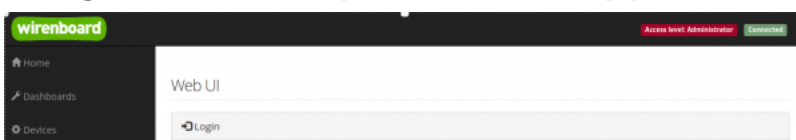
File	Title	Description
/etc/network/interfaces	Network Interface Configuration	Specifies network configuration of the system
/etc/ntp.conf	NTP configuration	Specifies NTP network time sync daemon configuration
/etc/wb-hardware.conf	Hardware Modules Configuration	Lists additional hardware modules configuration
/etc/wb-knxd-config.conf	KNXD Configuration	
/etc/wb-mqst-adc.conf	ADC Driver Configuration	Analog inputs configuration
/etc/wb-mqst-dac.conf	Analog Outputs Configuration	
/etc/wb-mqst-eb.conf	MQTT History Configuration	
/etc/wb-mqst-gpio.conf	GPIO Driver Configuration Type	Digital inputs and outputs configuration
/etc/wb-mqst-mbgate.conf	MQTT to Modbus TCP and RTU slave gateway configuration	
/etc/wb-mqst-opcu.conf	MQTT to OPC UA gateway configuration	Configure topics to fields mapping and daemon configuration
/etc/wb-mqst-serial.conf	Serial Device Driver Configuration	
/etc/wb-rules/alarms.conf	Alarm Configuration	Lists alarms
/etc/wb-webui.conf	WebUI Configuration	For internal use, edit with caution

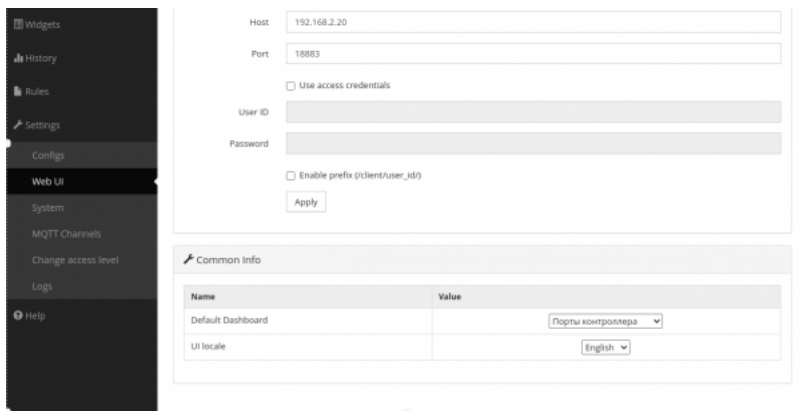
Страница Configs

На странице **Settings -> Configs** производится конфигурирование контроллера и настройка подключения внешних устройств:

- настройка сетевых интерфейсов
- настройка серверов получения точного времени
- конфигурирование и настройка боковых и внутренних модулей
- настройка сервиса knxd
- настройка аналоговых входов
- настройка записи в историю
- настройка цифровых входов и выходов (GPIO): в последних версиях контроллера список GPIO по умолчанию пустой, все входы-выходы сконфигурированы системой. Изменять назначение входов-выходов следует, если вы хотите изменить их режим функционирования. Список номеров GPIO для последних версий контроллеров Wren Board 6 представлен на странице [Подробное_тех.описание_платы_контроллера](#).
- настройка шлюза Modbus TCP / Modbus RTU
- настройка шлюза OPC UA
- настройка подключения устройств RS-485
- настройка предупреждений (alarms)
- доступ к редактированию JSON-файла настроек веб-интерфейса

Settings -> WebUI (Настройки -> Веб-интерфейс)



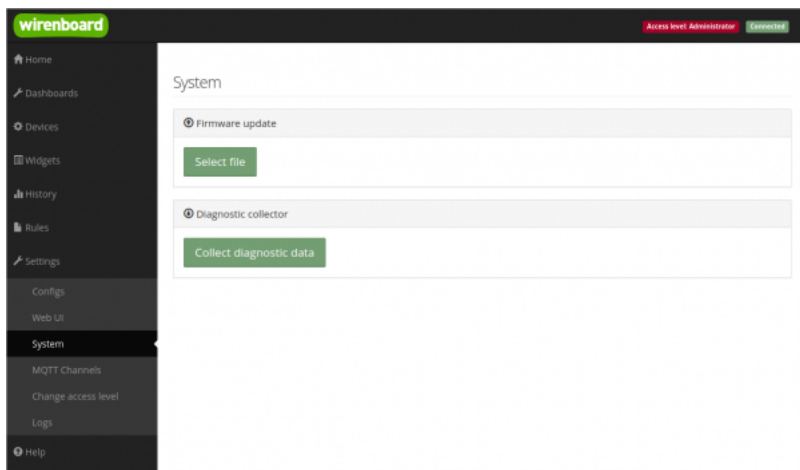


Страница Settings

На странице **Web UI** настраиваются параметры веб-интерфейса и контроллера. Здесь можно:

- Выбрать подключение к MQTT-брокеру (Web-sockets), если используется нелокальный брокер, а, например, облачный сервис
- При необходимости указать учетные данные на удаленном MQTT-брокере
- Указать префикс всех топиков, с которым данные охраняются в облачном сервисе
- Выбрать панель (Default Dashboard), которая будет отображаться на главной странице (Home)
- Выбрать язык веб-интерфейса.

Settings -> System (Настройки -> Системные)

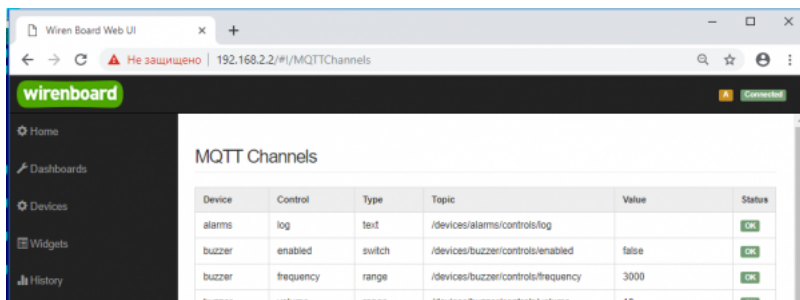


Страница System

На этой странице можно:

- Обновить прошивку контроллера, предварительно скачав ее на компьютер.
- Загрузить архив с диагностической информацией для отправки в техподдержку.

Settings -> MQTT Channels (Настройки -> MQTT-каналы)

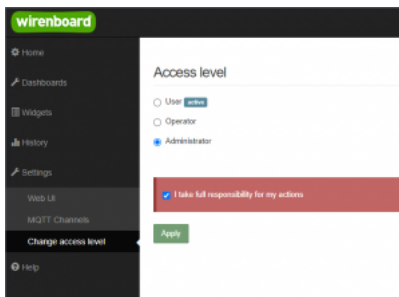


name	type	unit	path	value	status
hwmmon	Board Temperature	temperature	/devices/hwmmon/controls/Board Temperature	34.875	OK
hwmmon	CPU Temperature	temperature	/devices/hwmmon/controls/CPU Temperature	57.778	OK
network	Ethernet 2 IP	text	/devices/network/controls/Ethernet 2 IP	192.168.2.2	OK
network	Ethernet IP	text	/devices/network/controls/Ethernet IP		OK
network	GPRS IP	text	/devices/network/controls/GPRS IP		OK
network	Wi-Fi 2 IP	text	/devices/network/controls/Wi-Fi 2 IP		OK
network	Wi-Fi IP	text	/devices/network/controls/Wi-Fi IP	192.168.42.1	OK
power_status	Vin	voltage	/devices/power_status/controls/Vin	11.91	OK

MQTT Channels

На этой странице приводится справочная информация о всех MQTT-топиках, полученных веб-интерфейсом контроллера, а также статус их получения (**OK** или **ERR** в последнем столбце).

Settings -> Change access level (Настройки -> Права доступа)



Web UI 2.0 — смена уровня доступа текущего пользователя

Уровни доступа призваны защитить пользователя от ошибок при регулярной работе с контроллером. Важно понимать, что это не полноценное разграничение прав, а способ защитить себя от необдуманных действий. Новых пользователей создавать нельзя.

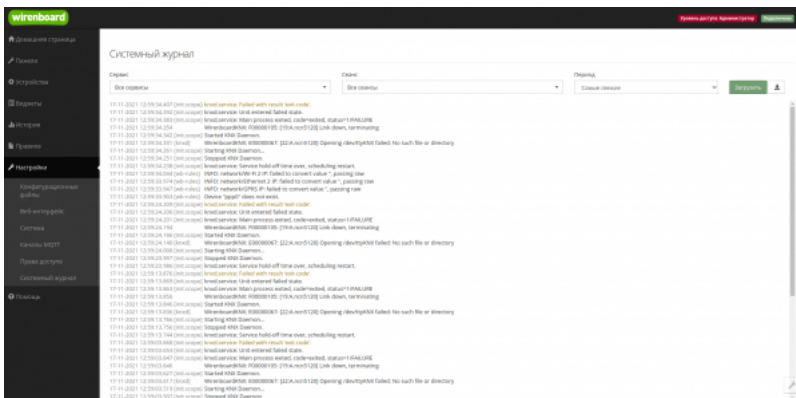
Доступны следующие уровни:

- User — дашборды, виджеты, история, базовые настройки.
- Operator — права уровня User и раздел Devices.
- Administrator — полный доступ ко всем функциям.

Чтобы изменить уровень доступа к настройкам веб-интерфейса:

- Зайдите в веб-интерфейс.
- Перейдите в раздел **Settings** и выберите пункт **Change access level**. Установите нужное значение и нажмите кнопку **Apply**.

Settings -> Logs (Настройки -> Логи)



Просмотр лог-файлов контроллера

Чтобы открыть инструмент, перейдите **Settings** → **Logs**.

После выбора параметров, нажмите **Load** для запроса данных из системного журнала.

Загруженные сообщения можно сохранить в файл, для этого нажмите кнопку **Save loaded log to file** и укажите куда

сохранить. Строки из системного журнала подгружаются во время прокрутки списка сообщений, поэтому сперва прокрутите список до нужного места, а потом сохраняйте вывод в файл.

Для отправки сообщений системного журнала в техподдержку удобно использовать выгрузку диагностической информации.

Стандартные задачи, решаемые через веб-интерфейс

Подключить устройство RS-485 Modbus и создать кнопки управления на главной панели

RS-485:Настройка через веб-интерфейс

Обновить прошивку контроллера

Обновление прошивки через веб-интерфейс

Облачный интерфейс

Веб-интерфейс Wiren Board можно разместить не только на самом контроллере, но и на специальном сервере. Тогда на интерфейс можно будет заходить, используя всегда один и тот же IP-адрес.

Чтобы контроллер начал работать с веб-интерфейсом, размещённым на сервере, нужно внести некоторые изменения в конфигурацию контроллера.

Такой вариант удобен, если ваш контроллер находится за роутером и не имеет глобального IP-адреса, или если он подключён по GPRS - тогда он тоже, скорее всего, не имеет глобального IP, да ещё и работа с удалённым веб-интерфейсом израсходует слишком много трафика.

Пока что такой вариант доступен только корпоративным клиентам по запросу.

Настройка авторизованного доступа к веб-интерфейсу контроллера

В статье Защита паролем приводятся краткие инструкции по перенастройке контроллера, обеспечивающие авторизованный доступ к веб-интерфейсу контроллера.

Обновление веб-интерфейса

Новые контроллеры поставляются с веб-интерфейсом версии 2.x.

Для обновления веб-интерфейса с предыдущих версий, нужно сделать:

```
apt update
apt install wb-mqtt-homeui
```

Проверьте установленную версию:

```
dpkg -s wb-mqtt-homeui
```

После установки зайдите через браузер в веб-интерфейс и одновременно нажмите клавиши Ctrl+Shift+R — это удалит страницу из кэша браузера и позволит избежать возможных проблем.

Основные отличия версии 2.x от 1.0

- Каждый виджет может содержать произвольное число каналов, в виджете каналы можно переименовывать
- Отдельные устройства теперь автоматически сворачиваются в виде плиток, если не помещаются на экране. Плитки можно развернуть или свернуть
- Появились уровни доступа к интерфейсу (пользователь, оператор, администратор). Текущий уровень доступа отображается правом верхнем углу интерфейса, рядом со значком состояния подключения
- Улучшенный интерфейс для мобильных устройств
- По клику на канал или значение название канала или его значение копируются в буфер обмена
- Историю значений можно посмотреть, нажав на кнопку, появляющуюся рядом со значением при наведении
- Историю значений можно скачивать в виде текстового файла

- исторические данные загружаются постепенно; возможно сравнивать значения нескольких каналов
- Удаление лишних MQTT-топиков из интерфейса
- Все настройки отображения теперь хранятся в конфиг-файле /etc/wb-ui.conf в формате JSON. Теперь их можно редактировать и генерировать из сторонних программ и очень просто копировать с одного контроллера на другой
- Отсутствуют "Комнаты"
- Сохранение конфигурации интерфейса при обновлении предыдущей версии веб-интерфейса.

wireboard

WB-MRWL3 Modbus Relay Modules

wireboard

https://wireboard.com/wiki/WB-MRWL3_Modbus_Relay_Module
28-04-2022 18:11

Модуль Modbus-реле WB-MRWL3

Руководство по эксплуатации

Самая актуальная документация всегда доступна на нашем сайте по ссылке: https://wireboard.com/wiki/WB-MRWL3_Modbus_Relay_Module

Этот документ составлен автоматически из основной страницы документации и ссылок первого уровня.

Содержание

[WB-MRWL3 Modbus Relay Modules](#)

[WB-MRWL3 Modbus Relay Module](#)

[Рекомендации по выбору реле](#)

[Mapping-матрица](#)

[Утилита «modbus_client»](#)

[Работа с Modbus-устройствами Wiren Board без контроллера](#)

[Протокол Modbus](#)

[RS-485](#)

[Веб-интерфейс Wiren Board](#)

[Настройка параметров подключения по RS-485 для Modbus-устройств Wiren Board](#)

[Modbus-адрес устройства Wiren Board](#)

[Карта регистров модулей реле](#)

[Обновление прошивки Modbus-устройств Wiren Board](#)

[Журнал изменений прошивок](#)

WB-MRWL3 Modbus Relay Modules

- [English](#)
- [русский](#)

[Купить в интернет-магазине](#)

Contents

[Назначение](#)

[Технические характеристики](#)

[Общий принцип работы](#)

[Входы](#)

[Выходы](#)

[Параметры реле](#)

[Монтаж](#)

[Пример монтажа](#)

[Настройка](#)

[Способы настройки](#)

[Режим работы реле при возобновлении питания](#)

[Безопасный режим](#)

[Режимы взаимодействия входов и реле](#)

[Антидребезг](#)

[Представление в веб-интерфейсе контроллера WB](#)

[Выбор шаблона](#)

[Управление устройством и просмотр значений](#)

[Работа по Modbus](#)

[Параметры порта по умолчанию](#)

[Modbus-адрес](#)

[Карта регистров](#)

[Обновление прошивки и сброс настроек](#)

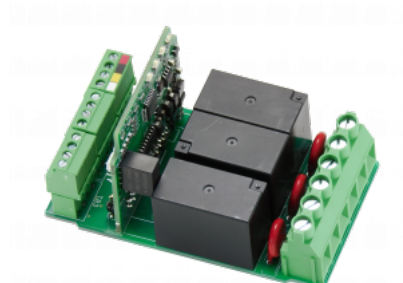
[Известные неисправности](#)

[Ревизии устройства](#)

[Изображения и чертежи устройства](#)



Релейный модуль WB-MRWL3



Плата WB-MRWL3

Назначение

Предназначены для коммутирования мощной силовой нагрузки общего назначения, в т.ч. индуктивной: управления группами розеток, двигателями, обогревателями и т.д.

Реле WB-MRWL3 рассчитаны на номинальную нагрузку до 20А (4,5 кВт) на канал и могут использоваться напрямую, без промежуточных контакторов, для коммутации розеточных групп и других нагрузок, защищённых автоматическим выключателем номиналом 16А.

О выборе модуля реле читайте в статье [Рекомендации по выбору реле для нагрузки](#).

Технические характеристики

Параметр	Значение
Питание	
Напряжение питания	12 — 28 В постоянного тока
Потребляемая мощность	<ul style="list-style-type: none"> ▪ В режиме холостого хода (со всеми выключенными реле) — 0,5 Вт ▪ Со всеми включенными реле — 2 Вт ▪ Пиковое значение — до 5 Вт в течение 20 мс
Выходы	
Тип выходов	Контакты механического реле
Конфигурация выходов	3 выхода, соединенных с контактами реле
Конфигурация контактов каждого выхода	Двухпозиционные, нормально открытые (С и NO)
Максимальное коммутируемое напряжение, АС	250 В
Максимальное коммутируемое напряжение, DC	30 В
Номинальный ток на канал	20 А
Сопротивление контактов	< 100 мОм
Напряжение изоляции между контроллером и выходом	1500 В (среднеквадратичное значение)
Срок жизни	см. таблицу в разделе #Параметры реле
Входы	
Количество входов (Inputs 0—3)	4 (3+1)

Тип входов	«Сухой контакт», групповая изоляция Напряжение на входе ~12 В. Ток при замыкании входа ~2 мА
Допустимое напряжение	от -20 В до +40 В
Частота и длительность импульсов	<ul style="list-style-type: none"> До 10 Гц (Т > 50 мс) - по умолчанию До 1 кГц при уменьшении времени защиты от дребезга До 5 кГц (с версии прошивки 1.15.0) (смотри таблицу ревизий)
Функции	<ul style="list-style-type: none"> Входы общего назначения Счет сигналов Измерение частоты (с версии прошивки 1.15.0) Прямое управление каналами реле Одновременное отключение всей нагрузки Гибкая настройка взаимодействия с реле с помощью mapping-матрицы (с версии прошивки 1.9.0)
Индикация	
Индикация питания и обмена данными	Зеленый светодиод Status (под верхней этикеткой)
Индикация состояния каналов реле	Красно-оранжевые светодиоды 1, 2.. (под верхней этикеткой)
Управление	
Интерфейс управления	RS-485
Изоляция интерфейса	Неизолированный
Протокол обмена данными	Modbus RTU, адрес задается программно, заводские настройки указаны на наклейке
Параметры интерфейса RS-485	Задаются программно, по умолчанию: скорость — 9600 бит/с; данные — 8 бит; бит чётности — нет (N); стоп-биты — 2
Готовность к работе после подачи питания	~2 с
Условия эксплуатации	
Температура воздуха	От -40 до +80 °С
Относительная влажность	До 92 %, без конденсации влаги
Клеммники и сечение проводов	
Рекомендуемое сечение провода с НШВИ, мм ²	0,75 — 1 для входов управления, 2 x 2,5 — для релейных выходов
Длина стандартной втулки НШВИ, мм	8 (10 для релейных выходов)
Момент затяжки винтов, Н•м	0,2 для входов управления, 0,4 — для силовых выходов
Габариты	
Ширина, DIN-юнитов	3
Габаритные размеры (Д x Ш x В)	53 x 90 x 58 мм
Масса (с коробкой)	175 г

Общий принцип работы

Входы

Входы выведены на разъемные клеммники и работают по принципу "сухой контакт". Кнопки или выключатели подключаются между входами блока и iGND. Можно использовать кнопки с фиксацией или без нее. Есть дополнительный вход «0» — по умолчанию настроен на отключение всех реле.

Дискретные входы можно использовать для прямого управления каналами реле или настроить внутреннюю логику взаимодействия входов с выходами. Подробнее смотрите в разделе [Режимы взаимодействия входов и реле](#). Реле обычно применяются для подключения настенного клавишного выключателя, чтобы управлять освещением напрямую.

С версии прошивки 1.12.0 изменился режим работы с выключателями — теперь по умолчанию модуль настроен на выключатели с фиксацией — состояние контактов реле повторяют состояние контактов выключателя. До этого модули были по умолчанию настроены на выключатели без фиксации — каждое замыкание входа меняло состояние реле на противоположное.

Для каждого входа можно изменить параметр времени антидребезга. Подробнее смотрите в разделе [Антидребезг](#).

На каждом канале доступно измерение количества срабатываний, а начиная с версии прошивки 1.15.0 можно измерять и частоту импульсов на входе.

Максимальное значение измеряемой частоты сигнала зависит от ревизии устройства и от количества измеряемых частотных сигналов. На испытании проводилось измерение шести сигналов одновременно на частоте 2.5 кГц.

Выходы

WB-MRWL3 имеет три независимых нормально-открытых выхода реле.

Реле сертифицированы только для использования на переменном токе, нагрузочная способность на постоянном токе может быть ниже. Каждый контакт реле защищен от перенапряжения варистором. В таблице ниже перечислены технические характеристики реле, используемых в WB-MRWL3.

Параметры реле

В таблице ниже указаны параметры для используемых реле, номинальный и максимальный токи отличаются для готового устройства, смотрите их в таблице [Технические характеристики](#).

Номинальный коммутируемый ток, резистивная нагрузка 230 В AC	20 А
Максимальный коммутируемый ток, резистивная нагрузка 230 В AC	26 А
Максимальный допустимый пусковой ток (в течение 20 мс, для контакта NO)	31 А
Сопротивление контактов	< 100 мОм
Материал контактов	AgSnO ₂
Конфигурация контактов	SPST (NO, COM)
Срок жизни: количество переключений для нагрузки 20 А / 230 В AC, cos φ = 1	30 000
Срок жизни: количество переключений без нагрузки	1 000 000
Модель реле	Hongfa HF161F-W
Особенности	Большой ток
Применение	Силовая нагрузка общего назначения, индуктивная нагрузка

Монтаж

Устройство монтируется на стандартную DIN-рейку шириной 35 мм. Клеммный блок «V+ GND A B» с шагом 3.5 мм служит для подключения питания и управления по шине RS-485. Для стабильной связи с устройством важно правильно организовать подключение к шине RS-485, читайте об этом в статье [RS-485:Физическое подключение](#).

Рекомендуем для монтажа использовать гибкие многожильные провода с обжатием концов втулочными наконечниками (НШВИ — наконечник штыревой втулочный изолированный).

При снятии изоляции провод должен зачищаться ровно по длине гильзы (можно зачистить больше, а потом откусить выступающий излишек). Для обжима (опрессовывания) используйте пресс-клещи (кримпер, «обжимка»). При монтаже обжатый наконечником провод не разрушается винтовым зажимом и надежно фиксируется.

Не прикладывайте чрезмерное усилие при завинчивании клеммы — это приводит к разрушению винтового разъема.

При включении реле модуль импульсно потребляет большую мощность (см. потребление). Используйте блок питания с запасом по мощности. При питании по длинному кабелю учитывайте падение напряжения на нем.

Срабатывание дискретных входов происходит при их замыкании на землю iGND. Возникающий ток невелик (~2 мА), так что тип кнопки или выключателя может быть любым. Также не играет роли длина и сечение кабеля. Но для предотвращения наводок от близко расположенных силовых линий лучше использовать витую пару и подключать землю iGND ко второй жиле пары.

Сечение проводов, подключаемых к винтовым зажимам выходов реле, должно соответствовать мощности коммутируемой нагрузки. Винтовые зажимы принимают провод сечением 2.5–4 мм². Выбрать правильное сечение провода для подключения нагрузки поможет таблица [Допустимый длительный ток для проводов и шнуров](#).

Модуль реле необходимо устанавливать таким образом, чтобы удовлетворять требованиям электробезопасности и не допускать случайного касания контактов, находящихся под высоким напряжением. Устройство должно эксплуатироваться при рекомендованных условиях окружающей среды.



Как обжимать наконечники НШВИ

Пример монтажа

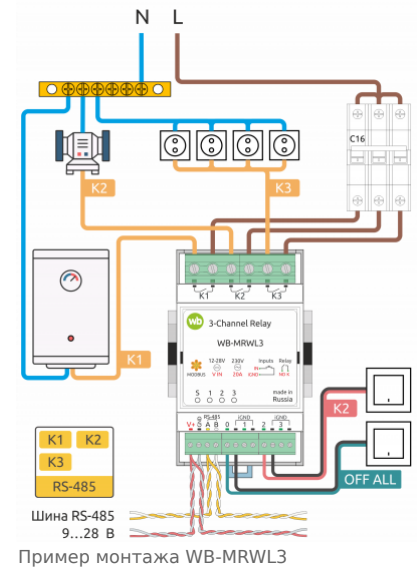
Один из вариантов подключения нагрузки к модулю WB-MRWL3 можно посмотреть на рисунке **Пример монтажа WB-MRWL3**.

В примере проводка до и после реле защищена автоматом на 16 А. Номинал выбирается с учетом сечения использованных при монтаже проводов и максимального коммутируемого модулем тока.

К выходам K1, K2, K3 подключена нагрузка: водонагреватель, электродвигатель и группа розеток.

Выход K2 может управляться как выключателем, так и по шине RS-485. Выходы K1 и K3 — только по шине RS-485. Выключатель OFF ALL обесточивает все выходы модуля реле K1 – K3. Назначение входов можно изменить с помощью [Mapping-матрицы](#).

Подробнее о выборе типа выключателей и других настройках модуля читайте в разделе [Настройка](#).



Настройка

Способы настройки

1. Указать параметры в веб-интерфейсе контроллера Wiren Board. Перейдите на [страницу настройки serial-устройств](#), выберите порт, найдите или добавьте устройство и измените параметры. Если нужный параметр отсутствует в шаблоне, его можно задать через [пользовательские параметры](#).
2. Записать настройки в [Modbus-регистры](#) модуля из консоли контроллера с помощью утилиты [modbus_client](#).
3. Если нет контроллера Wiren Board, используйте [адаптер USB-RS485](#).

Режим работы реле при возобновлении питания

Устройство запоминает состояние выходов при отключении питания, но вы можете это изменить.

Выберите нужный режим в параметре **Restore Last Outputs State After Power On**.

Безопасный режим

Безопасный режим позволяет остановить технологические процессы в случае, если контроль над модулем утрачен, например, перебит кабель RS-485. Рекомендуем учитывать возможность потери связи с контроллером при проектировании систем управления.

Таймер этого режима начинает отсчёт после каждого успешно обработанного (принятого) пакета Modbus. При достижении установленного времени — выходы реле отключаются. Если была включена маппинг-матрица, то реле можно управлять напрямую от его входов.

Значение таймера указывается в параметре **Safety Timer (s)** — значение «0» отключает безопасный режим. По умолчанию безопасный режим выключен.

Режимы взаимодействия входов и реле

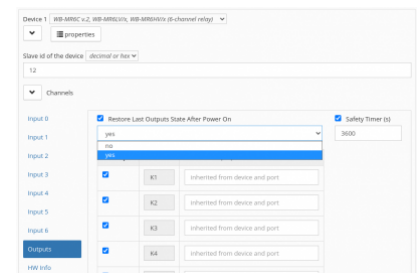
В модулях для каждого дискретного входа можно настроить внутреннюю логику, которая позволяет управлять выходами реле. Изменить режим можно в параметре **Input x Mode** или настроить логику через [Mapping-матрицу](#).

Начиная с версии прошивки 1.17.0 появились дополнительные матрицы с возможностью настроить входы как кнопки для детектирования различных типов нажатий: одинарное, двойное, длительное и т.п.

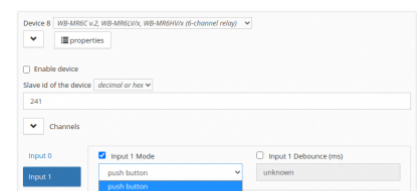
Кроме этого, можно полностью отключить обработку состояния любого из входов, притом, в веб-интерфейсе и регистрах можно будет отслеживать их состояние и обрабатывать программно на контроллере.

Режимы по умолчанию:

- каждый вход управляет соответствующим реле, т.е. вход номер 2 управляет реле



Пример настройки реле WB-MRWL3 v.2: таймер безопасного режима и выбор состояния выходов при возобновлении питания



каждый вход управляет соответствующим реле, т.е. вход номер 2 управляет реле номер 2

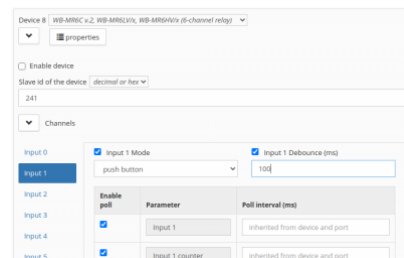
- режим работы для входов — выключатель с фиксацией (до середины 2019 года, кнопка без фиксации)
- нулевой вход отключает все реле.



Пример выбора режима для входа 1 реле WB-MR6C v.2 в веб-интерфейсе контроллера Wiren Board

Антидребезг

Для любого из входов реле вы можете настроить фильтр антидребезга в параметре **Input x Debounce (ms)**. Возможные значения от 0 до 100 мс, значение по умолчанию — 50 мс.



Пример установки времени антидребезга для входа 1 реле WB-MR6C v.2 в веб-интерфейсе контроллера Wiren Board

Представление в веб-интерфейсе контроллера WB

Выбор шаблона

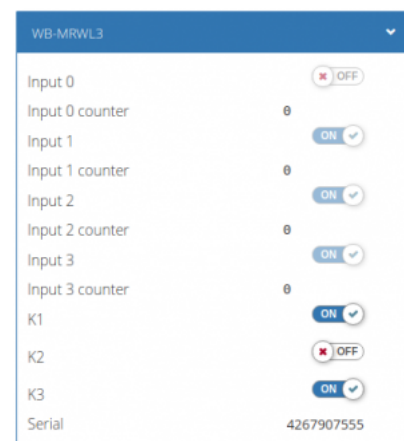
Чтобы устройство появилось на вкладке *Devices* в веб-интерфейсе контроллера Wiren Board, добавьте новое serial-устройство и выберите один из шаблонов:

- в текущем релизе — **WB-MR3LV/x**, **WB-MR3HV/x**, **WB-MRWL3**,
- в старых версиях ПО — **WB-MR3**.

Управление устройством и просмотр значений

В веб-интерфейсе вы можете управлять выходами устройства и просматривать полученные с него значения. Список отображаемых каналов можно изменить через настройки устройства, доступные на [странице выбора шаблона](#).

Через веб-интерфейс вы можете управлять выходами *K1-K3* и наблюдать состояние входов *Input 0 - Input 3*. В полях *Input N counter* указано значение счетчика срабатываний, а в поле *Serial* — серийный номер устройства.



Элементы управления и индикации модуля WB-MRWL3 в веб-интерфейсе

Внешний вид модуля в веб-интерфейсе изображен на рисунке.

Работа по Modbus

Устройства Wiren Board управляются по протоколу Modbus RTU. На физическом уровне подключаются через интерфейс [RS-485](#).

Поддерживаются все основные команды чтения и записи одного или нескольких регистров. Смотрите список доступных команд в [описании протокола Modbus](#).

Настроить параметры модуля можно в [веб-интерфейсе](#) контроллера Wiren Board, или через [сторонние программы](#).

Параметры порта по умолчанию

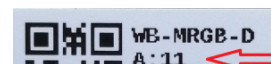
Значение по умолчанию	Название параметра в веб-интерфейсе	Параметр
9600	Baud rate	Скорость, бит/с
8	Data bits	Количество битов данных
None	Parity	Бит чётности
2	Stop bits	Количество стоповых битов

При необходимости их можно изменить, смотрите инструкцию в статье [Настройка параметров обмена данными](#).

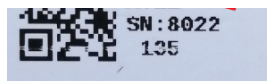
Для ускорения отклика устройств рекомендуем поднять скорость обмена до 115 200 бит/с.

Modbus-адрес

Каждое устройство на линии имеет уникальный адрес в диапазоне от 1 до 247. Адрес устройства, установленный на заводе, указан на отдельной наклейке со штрихкодом. На заводе устройствам



Wiirenboard в одной партии присваиваются разные адреса, поэтому в вашем заказе, скорее всего, адреса не будут повторяться.



Modbus-адрес,
установленный на
производстве

О том, как узнать, изменить или сбросить Modbus-адрес устройства, читайте в статье [Modbus-адрес устройства Wiiren Board](#).

Карта регистров

Карта регистров модулей реле

Обновление прошивки и сброс настроек

Большинство устройств Wiiren Board поддерживают обновление прошивки (микропрограммы) по протоколу Modbus. Это даёт возможность расширять функциональные возможности устройств и устранять ошибки в микропрограмме непосредственно на месте монтажа.

Инструкции:

- [Обновление прошивки](#)
- [Настройка параметров подключения](#)
- [Modbus-адрес: узнать, сбросить или изменить](#)

Узнать о выходе новой версии прошивки можно в [Журнале изменений прошивок](#).

Известные неисправности

[Список известных неисправностей](#)

Ревизии устройства

Номер партии (Batch №) указан на наклейке на боковой поверхности корпуса или на печатной плате.

Ревизия	Партии	Дата выпуска	Отличия от предыдущей ревизии
1.3	v1.3A - v1.3C - ...	11.2021 - ...	<ul style="list-style-type: none">▪ на микроконтроллерной плате новой ревизии
1.2	v1.2Q	10.2021 - 11.2021	<ul style="list-style-type: none">▪ без конденсаторов на входах (улучшено быстродействие входов)
1.2	v1.2N, v1.2O/1, v1.2O/2, v1.2P	06.2021 - 10.2021	<ul style="list-style-type: none">▪ на микроконтроллере GD32
1.2	v1.2O	05.2021 - 06.2021	<ul style="list-style-type: none">▪ на микроконтроллере STM32F042K6T6
1.2	v1.2C - v1.2M	12.2019 - 05.2021	<ul style="list-style-type: none">▪ напряжение на входах типа "сухой контакт" стало ~12В▪ увеличена максимальная частота работы входов до 1 кГц (для прошивок начиная с 1.13.0)
1.2	v1.2A - v1.2B	01.2019 - 11.2019	<ul style="list-style-type: none">▪ с разъёмными клеммниками DEGSON
1.1	241, MRLW3/1	до 03.2019	<ul style="list-style-type: none">▪ напряжение на входах типа "сухой контакт" ~4,5В▪ с разъёмными клеммниками KEFA

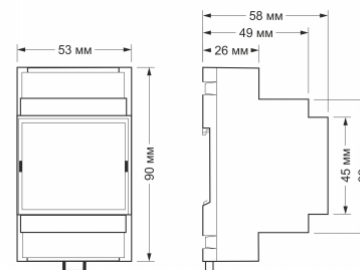
Изображения и чертежи устройства

Corel Draw 2018 (шрифт — Ubuntu): [Файл:WB-Library.cdr.zip](#)

Corel Draw PDF: [Файл:WB-MRWL3.cdr.pdf](#)

Autocad 2013 DXF: [Файл:WB-MRWL3.dxf.zip](#)

Autocad PDF: [Файл:WB-MRWL3.pdf](#)



Габаритные размеры WB-MRWL3 составляют 53,3 x 90,2 x 57,5 мм (Д x Ш x В)

WB-MRWL3 Modbus Relay Module

- English
- русский

Buy online

Three-channel relay modules WB-MRWL3 are used in industrial and home automation systems. They are intended for commutation of powerful power load of General purpose, including inductive: control of groups of sockets, engines, heaters, etc.

WB-MRWL3 modules are designed for rated load up to 20A (4.5 kW) per channel and can be used directly without intermediate contactors for switching outlet groups and other loads protected by an automatic switch with a nominal value of 16A.

WB-MRWL3 has an input that simultaneously deactivates all the load.

Contents

Outputs

Relay parameters



Inputs

Technical specifications

[Габаритные размеры модулей](#)

Installation of modules

Relay operation in case of power failure

Modbus Control

Device firmware update

Known faults and bugs

Images and drawings of the device



Outputs

The module has three independent normally-open relay outputs.

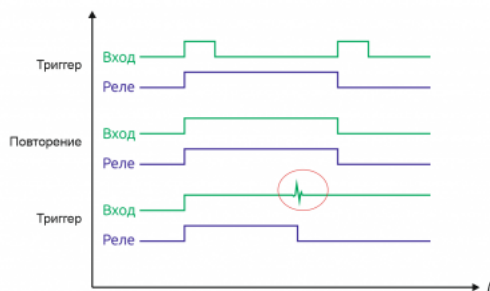
Relays are certified only for use on alternating current, the load capacity of the DC may be lower. Each relay contact is protected from arcing when the varistor closes/opens. Depending on the Modbus register settings, the state of the inputs can be resumed after a power interruption (see [#Relay operation mode in case of power loss](#)). The table below lists the technical characteristics of the relays used in the module.

Relay parameters

Our detailed article in Russian on [habr.com](https://habr.com/ru/company/wirenboard/blog/422197/) about the choice of relay parameters for the switched load: <https://habr.com/ru/company/wirenboard/blog/422197/>

Maximum switching current, resistive load 230V AC	20A
Maximum allowable inrush current (within 20ms, for no contact)	30A
Contact resistance	< 100 milliohms
Contact material	AgSnO ₂
Contact configuration	SPST (NO, COM)
Lifetime: number of switches for load 20A 230V AC, cos φ = 1	30 000
Lifetime: number of switches without load	1 000 000
Relay model	Hongfa HF161F-W
Peculiar properties	High current
Application	General purpose power load, inductive load.

Inputs



Closed input in trigger mode may be unstable

The inputs of the WB-MRWL3 modules are led out on detachable screw terminals. Inputs are organized on the dry contact principle. Buttons or switches are connected between the corresponding inputs and iGND inputs.

The 3 inputs can be used for direct control of the respective relay channels. Another input can be used for group disconnection of all channels.

The inputs have two modes of operation: trigger mode (for control with buttons) and repeat mode (for control with switches). In the trigger mode, to change the state of the relay, it is enough to close and open the input once. The change of state occurs on the rising edge (that is, when the circuit). In the repeat mode, the relay state is determined by the input state: the input is closed — the relay is on, open — off. Details about the mode switching written in [Modbus Management Of Relay Modules](#).

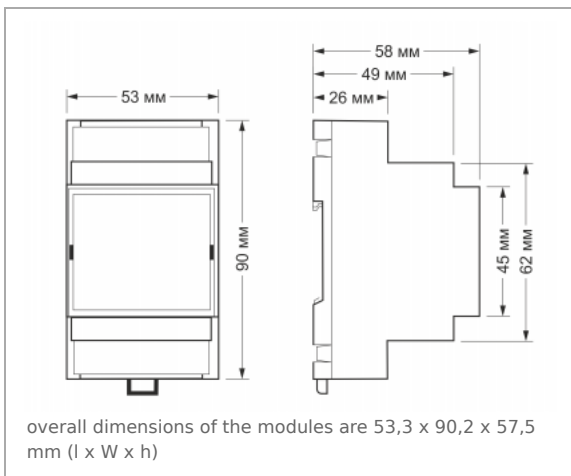
IMPORTANT: it is not recommended to use switches (lock buttons) in the trigger mode of inputs. Input interference may be perceived as a short release/push of the button and the relay status will change, although the switch will remain closed.

Technical specifications

Parameter	Value
Power	
Supply voltage	12 — 28 V DC
Power consumption	<ul style="list-style-type: none"> ▪ Idle mode (All relays switched off) — 0.5 W ▪ All relays switched on — 2 W ▪ Peak value — up to 5 W for 20 MS
Terminals and wire section	
Recommended wire cross-section with NSUI, mm ²	0,75 — 1 for control inputs, 2 x 2.5 for relay outputs
Length of the standard sleeve NSVE, mm	8 (10 for relay outputs)
Torque of screws, N•m	0,2 for control inputs , 0,4 — for power outputs
Outputs	
Output type	Mechanical relay contacts
Output configuration	3 / output connected to the relay contacts
Configuration of each output contacts	Two-position, normally open (S and NO)
Maximum switching voltage, AC	250 V
Maximum switching voltage, DC	30 Volts
Rated current per channel	20 A
Contact resistance	< 100 milliohms

Voltage isolation between the controller and the output	1500 V (RMS value)
Term life:	see table in #Relay parameters
Inputs	
Number of inputs (Inputs 0—3)	4 (3+1)
Input type	"Dry contact", group insulation. The input voltage is ~4.5 V. Current closing input ~2mA.
Functions	<ul style="list-style-type: none"> ▪ General purpose inputs ▪ Signal count ▪ Direct control of relay channels ▪ Simultaneous disconnection of the entire load ▪ Remember the relay status in case of power cuts ▪ Flexible setup of the interaction of inputs with relay with <u>mapping-matrix</u>, since firmware version 1.9.0
Control	
Management interface	RS-485
Interface isolation	Uninsulated
Communication protocol	Modbus RTU, the address is set by software, factory settings are indicated on the label
RS-485 interface parameters	set programmatically (see <u>register card</u>); default: speed — 9600 bit/s; data — 8 bit; parity bit — none (N); stop bits — 2
Ready for operation after power supply	~0.03 c
dimensions	
Width, DIN units	3
Overall dimensions (l x W x h)	53.3 x 90.2 x 57.5 mm
Display	
Power and data exchange indication	Green status led
Relay channel status indication	Red LEDs, 1 — 3
Operating conditions	
Air temperature	
Relative humidity	Up to 85%, without condensation

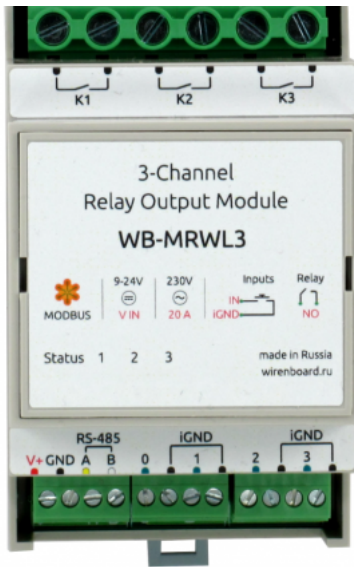
Габаритные размеры модулей



Installation of modules



The relay module is mounted on a standard 35 mm wide DIN rail and occupies a width of 3 DIN modules.



Relay module pinout

All connections are realized by blocks of screw clips. The wire section for contacts K1 — K3 must correspond to the switched load.

When connecting the switching devices to the relay contacts K1 — K3, it is important not to exceed the recommended current — 20 A.

The flowing current is small when the button or switch is closed, so the button or switch can be any type. Despite the built-in protection against rattle, choose quality buttons and switches to avoid false alarms due to bad contacts.

If the device is the last one on the RS-485 line, a 120 Ohm resistor-terminator must be installed between its inputs A and B. Practice shows that in the case of bench tests with a small length of the RS-485 line and a small number of terminator devices on the last device in the line can be installed without terminator.

The module must be installed in such a way as to meet the requirements of electrical safety and to prevent accidental contact of contacts under high voltage. The module must be operated under recommended environmental conditions.

Relay operation in case of power failure

Depending on the value written to the storage register 0x06, after the power is turned off and restored, the relays will remain off (if the value is 0, or turn on those that were turned on before the power interruption if the register has a value of 1).

Modbus Control

Details of the work with the module via Modbus Protocol is written in [Relay Modules Modbus Management](#) . There you can also find a map of the device registers.

[Back to peripheral devices list](#)

Device firmware update

Starting with the new firmware versions of the device, the firmware update functionality (firmware) via the Modbus Protocol is supported. This makes it possible to extend the functionality of the devices and eliminate errors in the firmware directly at the installation site. Read more about flashing devices described in the article [Download firmware to Wiren Board devices](#). A list of firmware versions that support the update, as well as devices for which this feature is implemented, can be found in [Firmware Changelog](#).

Known faults and bugs

[List of known faults](#)

Images and drawings of the device

По ссылкам ниже вы можете скачать изображения и чертежи устройства **WB-MRWL3**.

Corel Draw 2018: [WB-MRWL3.cdr.zip](#)

Corel Draw PDF: [WB-MRWL3.cdr.pdf](#)

Autocad 2013 DXF: [WB-MRWL3.dxf.zip](#)

Autocad PDF: [WB-MRWL3.pdf](#)

Рекомендации по выбору реле

Contents

[Сводная таблица](#)

[Лампы накаливания](#)

[Лампы светодиодные и компактные люминесцентные](#)

[Теплые полы. Чайник, ТЭНы электродуховок](#)

[Электродвигатели](#)

[Блоки питания](#)

[Выводы](#)

Сводная таблица

Тип нагрузки	Пусковой ток, длительность	Мощность группы	Допустимо	Рекомендуем
		<250Вт	WB-MR6C , WBIO-DO-R10A-8	MR6C v.2.

Лампы накаливания	8-12*I _{ном} , 5 мс	<1кВт		WB-MR6CU v.2
		<1,5кВт	WB-MR3LV/I или WB-MR6LV/I*	WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6
		<2кВт		
Светодиодные "хорошие"	10-20*I _{ном} , 100 мкс	<150Вт	WB-MR6C, R10A8	MR6C v.2, WB-MR6CU v.2
		<600Вт		
		<900Вт	WB-MR3LV/I или WB-MR6LV/I	WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6
		<2кВт		
Светодиодные "плохие" и компактные люминесцентные лампы	150-200*I _{ном} , 100 мкс	<15Вт	WB-MR6C, WBIO-DO-R10A-8	MR6C v.2, WB-MR6CU v.2
		<60Вт		
		<90Вт	WB-MR3LV/I или WB-MR6LV/I*	WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6
		<120Вт		
Электродвигатели	2-5*I _{ном} , 100 мс - 2 сек.	<300Вт	WB-MR6C, WBIO-DO-R10A-8	
		<500Вт	WB-MR3LV/I или WB-MR6LV/I*, WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6	WB-MR6C v.2, WB-MR6CU v.2
		<800Вт		WB-MRWL3
Импульсные блоки питания	200-600*I _{ном} , 200-1000 мкс	<120Вт		WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6
Эл. котлы, чайники, тёплые полы и другая резистивная нагрузка	1*I _{ном}	<1.5кВт	WB-MR6C, WBIO-DO-R10A-8	WB-MR6C, WBIO-DO-R10A-8
		<2кВт	WB-MR3LV/I или WB-MR6LV/I*	WB-MR6C v.2, WB-MR6CU v.2, WB-MR3LV/S или WB-MR6LV/S, WB-MRPS6
		<4кВт		WB-MRWL3
Группы розеток	??	<3кВт		WB-MRWL3

* для контакта NO, для NC допустимая мощность в ~3 раза меньше

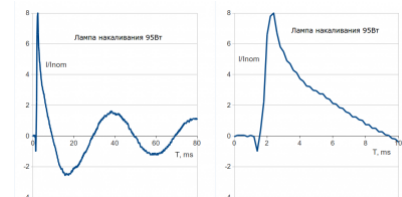
Методика расчета: Максимальный коммутируемый ток для реле делится на кратность пускового тока нагрузки, делится на 1,41, умножается на 230В, округляется до целого вниз. Или по мощности, если по нему ограничение меньше.

Почему именно так? Ведь в характеристиках используемых реле ток (а значит и мощность) гораздо выше рекомендуемых. Ответ прост - **пусковые токи**.

Разберем категории нагрузок.

Лампы накаливания

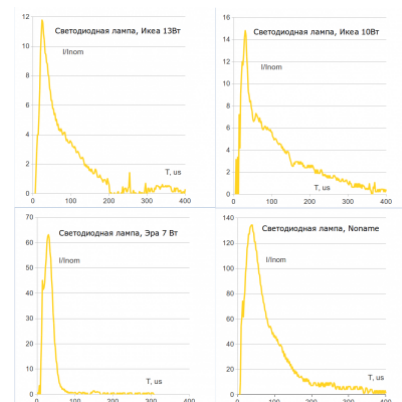
Принято считать, что это - исключительно резистивная нагрузка. Но пока спираль лампы холодная, она имеет сопротивление сильно меньшее по сравнению с рабочим. Лампа накаливания мощностью 95 Вт имеет сопротивление 40 Ом, что оценочно дает пусковой ток до 320 В / 40 Ом = 8 А, то есть, в 13 раз больше номинального. Видно, что пусковой ток превышает номинал в 8 раз, время разогрева спирали составляет менее одного полупериода, а длительность пика — примерно 2 мс.



Ток через спираль холодной лампы

Лампы светодиодные и компактные люминесцентные

Такие лампы небольшой мощности содержат в себе выпрямитель (диодный мост) с конденсатором. То есть, это чисто емкостная нагрузка, и пусковой ток должен быть очень большим. Как правило, для его снижения производители ставят перед мостом резистор и(или) термистор. Видно, что у ламп из IKEA всё весьма хорошо. А вот у других светодиодных ламп пусковой ток превышает номинал в 150 — 200 раз, и длительность пиков составляет ~100 мкс.



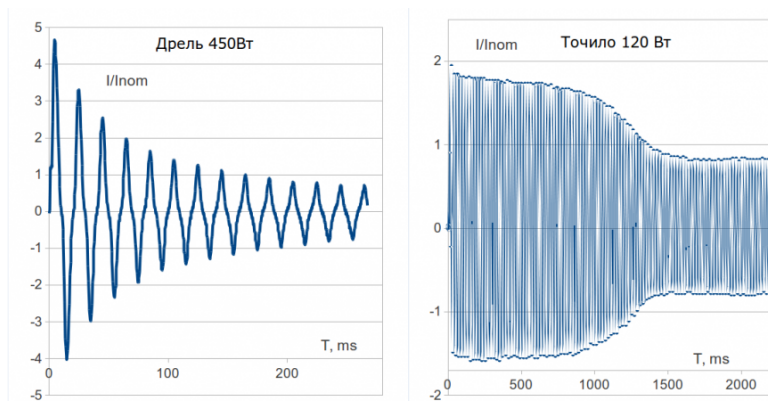
Ток включения энергосберегающей лампы

Тёплые полы. Чайник, ТЭНы электродкотлов

Температурный коэффициент нихромовых спиралей для рабочих температур в ТЭНах весьма мал, и пусковой ток близок к номинальному. Исключение — саморегулирующиеся кабели для теплых полов. У них - полупроводниковый нагревающий элемент, его пусковой ток может быть больше в 2 раза.

Электродвигатели

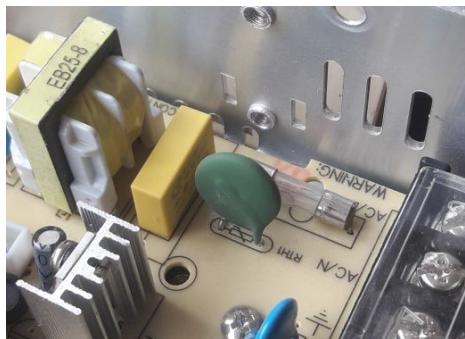
Верно, что у индуктивной нагрузки пусковой ток нулевой. И да, в момент замыкания контактов ток и правда нарастает плавно, но затем: 1. если момент замыкания попал в ноль напряжения, то всплеск тока двукратный (для чисто индуктивной нагрузки); 2. пока двигатель не выйдет на номинальные обороты, ток превышает в несколько раз номинальный; чем мощнее двигатель, тем больше превышение.



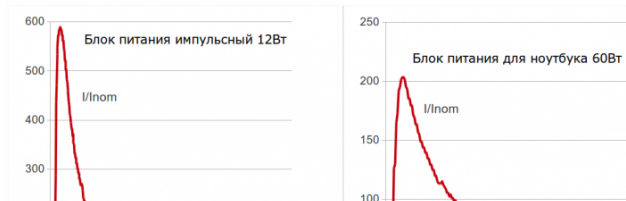
Ток включения двигателя

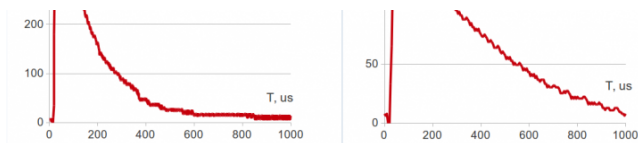
Блоки питания

Аналогично светодиодным лампам на входе у этих блоков питания стоит диодный мост и конденсаторы большой емкости. Для снижения пусковых токов производители ставят NTC-термисторы, зеленые (иногда черные) и круглые. В холодном состоянии они имеют заметное сопротивление, чем и ограничивают пусковой ток. При работе блока питания термистор нагревается, и его сопротивление снижается (в 20 — 30 раз), практически не мешая протеканию тока. Но после выключения блока питания некоторое время (до 1 минуты) термистор остается горячим и не может ограничивать пусковой ток. Поэтому крайне желательно после выключения блока питания подождать 10 — 30 с перед его повторным включением. Ниже представлены графики с повторным включением через ~15 с (при быстром переключении пики еще больше):



Термистор на плате БП





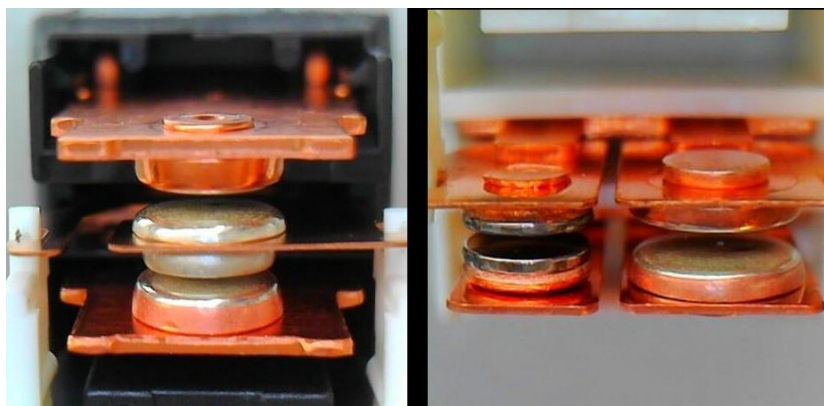
Ток включения БП

Выводы

В документации на реле могут указывать несколько токов: номинальный ток (Contact rating current) и максимальный ток переключения (Max. switching current) или пусковой ток (Inrush current) и т. д. И у «обычных» реле пусковой ток часто не указывают. То есть, если на реле написано «10А», то значит, по умолчанию у него и пусковой ток при коммутации не должен превышать 10А. Возможно, его можно умножить на 2, но это не точно.

Если максимальный пусковой ток 10-20А, а светодиодная лампочка имеет пусковой ток в 100 раз больше от номинала, то это очень грустно: получается, что коммутировать можно только 20-40 Вт лампочек. Так что с обычными реле нужно либо сильно ограничить себя в выборе нагрузки и занижать мощность, либо быть готовым к тому, что контакты будут часто свариваться, и реле придется менять. Для нагрузки с большими пусковыми токами лучше использовать специальные реле.

Отличие реле HF115F-I — особые контакты из AgSnO₂, а HF115F-S еще и имеют специальную конструкцию из двух пар контактов, когда первая пара (вольфрамовые контакты, большой импульсный ток) замыкается чуть раньше второй (низкое сопротивление контакта, большой постоянный ток).



Контакты реле HF115F-I (слева) и HF115F-S (справа)

Mapping-матрица

- [English](#)
- [русский](#)

Назначение

Релейные модули имеют несколько режимов работы реле от входов. И один из них — это взаимодействие согласно Mapping-матрице.

Режимы настраиваются индивидуально на каждый вход регистрами № 9-16, см. [карту регистров модулей реле](#).

Матрица действий над выходами по изменению состояний входов позволяет гибко настраивать реакцию устройства в зависимости от изменения его входов. Эта функциональность появилась с версии **1.9.0** для устройств серии **WB-MR** и **WB-MWAC**.

Устройство и принцип работы

Mapping-матрица находится в регистрах начиная с адреса 384, и

Регистры

Выходы MR3-6

содержит 64 регистра из расчета 8 входов / 8 выходов. В каждом регистре записан код действия входа на выход. Если в устройстве количество входов и/или выходов меньше 8, то адреса регистров **не** меняются — просто строки/столбцы для отсутствующих регистров не учитываются. Исключение — устройства с 2 входами и выходами.

	1	2	3	4	5	6	7	8
1	384	385	386	387	388	389	390	391
2	392	393	394	395	396	397	398	399
3	400	401	402	403	404	405	406	407
4	408	409	410	411	412	413	414	415
5	416	417	418	419	420	421	422	423
6	424	425	426	427	428	429	430	431
7	432	433	434	435	436	437	438	439
0	440	441	442	443	444	445	446	447

В ячейке на пересечении строки входа и столбца выхода — регистр действия входа на выход. В него записывается 16-битное слово **0b0000 0000 0000 ууxx**, где биты **уу** описывают действия при замыкании входа (переход из 0 в 1, передний фронт), а биты **xx** — при размыкании (из 1 в 0, задний фронт).

Действия, в зависимости от битов:

- 00 — ничего не делать
- 01 — выключить
- 10 — включить
- 11 — инвертировать значение

Регистры	Выходы MR2	
	1	2
Входы	1	384 385
	2	386 387

Таким образом, можно запрограммировать реакцию каждого выхода на замыкание и размыкание любых входов.

Примеры программирования взаимодействия входов и выходов

		Биты xx — размыкание входа			
		00	01	10	
Биты уу, замыкание входа	00	<p>(0) — Вход отключен, не управляет выходами</p>	<p>(1) — Выключить при размыкании</p>	<p>(2) — Включить при размыкании</p>	<p>(3) — Изме</p>
	01	<p>(4) — Выключить при замыкании</p>	<p>(5) — Всегда выключать</p>	<p>(6) — Работать как инвертированный выключатель с фиксацией</p>	<p>(7) — Выкл</p>
	10				

11	1000		(9) — Работать как выключатель с фиксацией (повторять вход)	1010		(11, 0x0B) — Всегда включать состояние
	1100		(12, 0x0C) — Изменить состояние выхода при замыкании (выключатель без фиксации)	1101		(13, 0x0D) — Изменить состояние, потом выключить при размыкании
						(14, 0x0E) — Изменить состояние, потом включить при размыкании
						(15, 0x0F) — и вернуться пр

Начиная с версии прошивки 1.17.0 появились дополнительные матрицы с возможностью настроить входы как кнопки для детектирования различных четырёх типов нажатий:

- короткое,
- длинное,
- двойное
- и сначала короткое, затем длинное.

Настраиваемые действия аналогичны Mapping-матрице, также как и принцип выбора регистра на пересечении. Биты **уу** не используются.

Карты регистров для настройки матриц нажатий описаны ниже, про настройку можно почитать в статье [Примеры программирования взаимодействия входов и выходов](#)

Mapping-матрица коротких нажатий:

Регистры	Выходы MR3-6								
	1	2	3	4	5	6	7	8	
Входы	1	544	545	546	547	548	549	550	551
	2	552	553	554	555	556	557	558	559
	3	560	561	562	563	564	565	566	567
	4	568	569	570	571	572	573	574	575
	5	576	577	578	579	580	581	582	583
	6	584	585	586	587	588	589	590	591
	7	592	593	594	595	596	597	598	599
	0	600	601	602	603	604	605	606	607

Регистры	Выходы MR2		
	1	2	
Входы	1	544	545
	2	546	547

Mapping-матрица длинных нажатий:

Регистры	Выходы MR3-6								
	1	2	3	4	5	6	7	8	
Входы	1	608	609	610	611	612	613	614	615
	2	616	617	618	619	620	621	622	623
	3	624	625	626	627	628	629	630	631
	4	632	633	634	635	636	637	638	639
	5	640	641	642	643	644	645	646	647
	6	648	649	650	651	652	653	654	655
	7	656	657	658	659	660	661	662	663
	0	664	665	666	667	668	669	670	671

Регистры	Выходы MR2		
	1	2	
Входы	1	608	609
	2	610	611

Mapping-матрица двойных нажатий:

Регистры	Выходы MR3-6							
	1	2	3	4	5	6	7	8

Входы	1	672	673	674	675	676	677	678	679
	2	680	681	682	683	684	685	686	687
	3	688	689	690	691	692	693	694	695
	4	696	697	698	699	700	701	702	703
	5	704	705	706	707	708	709	710	711
	6	712	713	714	715	716	717	718	719
	7	720	721	722	723	724	725	726	727
	0	728	729	730	731	732	733	734	735

Регистры	Выходы MR2	
	1	2
Входы	1	672 673
	2	674 675

Mapping-матрица сначала коротких, потом длинных нажатий:

Регистры	Выходы MR3-6								
	1	2	3	4	5	6	7	8	
Входы	1	736	737	738	739	740	741	742	743
	2	744	745	746	747	748	749	750	751
	3	752	753	754	755	756	757	758	759
	4	760	761	762	763	764	765	766	767
	5	768	769	770	771	772	773	774	775
	6	776	777	778	779	780	781	782	783
	7	784	785	786	787	788	789	790	791
	0	792	793	794	795	796	797	798	799

Регистры	Выходы MR2	
	1	2
Входы	1	736 737
	2	738 739

Утилита «modbus_client»

Contents

Описание

Подготовка к работе

Контроллер Wiren Board

Настольный компьютер с Linux

Аргументы командной строки

Примеры использования с оборудованием Wiren Board

Проверка подключения к устройству и считывание адреса

Запись нового адреса

Чтение сигнатуры устройства

Чтение версии прошивки

Настройка параметров трансформаторов

Включение реле релейного модуля

Одновременное включение нескольких реле

Настройка взаимодействия входов и выходов реле

Описание

modbus_client — утилита для опроса устройств по протоколам Modbus RTU и Modbus TCP из командной строки.

Подготовка к работе

Контроллер Wiren Board

Утилита `modbus_client` предустановлена на все контроллеры Wiren Board. Для использования утилиты нужно подключиться к контроллеру по протоколу SSH.

Обычно порт RS-485 занят драйвером `wb-mqtt-serial`, поэтому перед запуском `modbus_client` этот драйвер надо остановить:

```
service wb-mqtt-serial stop # для Wiren Board 5 и позднее
service wb-homa-modbus stop # для Wiren Board 4
```

После завершения работы с `modbus_client` запустите драйвер обратно:

```
service wb-mqtt-serial start # для Wiren Board 5 и позднее
service wb-homa-modbus start # для Wiren Board 4
```

Настольный компьютер с Linux

Скачайте пакет для настольных компьютеров с Linux (https://github.com/contactless/modbus-utils/releases/download/1.2/modbus-utils_1.2_amd64.deb).

Перейдите в папку со скаченным пакетом и установите его командой:

```
sudo apt install ./modbus-utils_1.2_amd64.deb
```

Также автоматически должен установиться пакет `libmodbus`, если этого не произошло — установите его из репозитория `apt`.

Аргументы командной строки

Значения параметров (адрес устройства или регистра, таймаут, тип функции, значение для записи в регистр и т.д.) можно указывать как в шестнадцатеричном `0x**`, так и в десятичном виде.

Вызов `modbus_client` без аргументов выдает краткое описание возможных аргументов команды:

```
modbus_client [--debug] [-m {rtu|tcp}] [-a<slave-addr=1>] [-c<read-no>=1]
[-r<start-addr>=100] [-t<f-type>] [-o<timeout-ms>=1000] [{rtu-params|tcp-params}] serialport|host [<write-data>]
NOTE: if first reference address starts at 0, set -0
f-type:
    (0x01) Read Coils, (0x02) Read Discrete Inputs, (0x05) Write Single Coil
    (0x03) Read Holding Registers, (0x04) Read Input Registers, (0x06) WriteSingle Register
    (0x0F) WriteMultipleCoils, (0x10) Write Multiple register
rtu-params:
    b<baud-rate>=9600
    d{7|8}<data-bits>=8
    s{1|2}<stop-bits>=1
    p{none|even|odd}=even
tcp-params:
    p<port>=502
Examples (run with default mbServer at port 1502):
Write data:  modbus_client --debug -mtcp -t0x10 -r0 -p1502 127.0.0.1 0x01 0x02 0x03
Read that data: modbus_client --debug -mtcp -t0x03 -r0 -p1502 127.0.0.1 -c3
```

Общие аргументы

Параметр	Описание	Обязателен	Значение по умолчанию
<code>--debug</code>	Может указываться в любой позиции и включает отладку, вывода на экран шестнадцатеричные коды отправляемых и принимаемых данных.	нет	
<code>-m</code>	Определяет тип используемого протокола: <ul style="list-style-type: none">▪ <code>-mrtu</code> — Modbus RTU,▪ <code>-mtcp</code> — Modbus TCP. Он должен указываться первым в командной строке, или вторым, если первый аргумент — <code>--debug</code> или имя файла порта RS-485.	да	
<code>-a</code>	Задаёт Modbus-адрес устройства, к которому мы обращаемся.	нет	1
<code>-c</code>	Определяет, какое количество элементов мы запрашиваем.	нет	1
<code>-r</code>	Задаёт начальный адрес для чтения или записи.	нет	100

-t	Указывает код функции Modbus. Кратко они перечислены в выводе modbus_client, подробнее значения кодов описаны на странице Протокол Modbus.	да	
-o	Задаёт таймаут в миллисекундах.	нет	1000
-0	Ноль. Уменьшает на единицу адрес, задаваемый аргументом -r. Это может быть полезным при работе с устройствами с нестандартной адресацией, например, с диапазоном адресов 1 — 65536 вместо привычного 0 — 65535.	нет	

Затем указываются специфические параметры протокола (Modbus RTU или Modbus TCP). Несмотря на информацию, выводимую в подсказке, эти параметры также начинаются со знака - (минус, дефис).

Для Modbus RTU

Параметр	Описание	Значение по умолчанию
-b	Скорость передачи данных по последовательной линии	9600
-d	Количество передаваемых бит данных, 7 или 8	8
-s	Количество стоповых битов, 1 или 2	1
-p	Контроль четности: <ul style="list-style-type: none"> -pnone — нет проверки, -peven — передается бит контроля на четность, -podd — передается бит контроля на нечетность. 	even

Для Modbus TCP

Параметр	Описание
-p	Номер TCP-порта устройства, с которым взаимодействует контроллер.

Далее следует имя файла порта RS-485 или адрес хоста, а в конце необязательный параметр — данные для функций записи.

Примеры использования с оборудованием Wiren Board

Проверка подключения к устройству и считывание адреса

Все устройства Wiren Board с протоколом Modbus RTU хранят адрес в регистре 128 — его удобно считывать для проверки подключения.

Читаем содержимое регистра 128 из устройства с адресом 2, подключенного к serial-порту /dev/ttyRS485-1, с помощью функции 0x03 (Read Holding Registers):

```
modbus_client --debug -mrtu -b9600 -pnone -s2 /dev/ttyRS485-1 -a2 -t0x03 -r128
```

Аргумент	Описание
--debug	отладка включена, будут выведены шестнадцатеричные коды отправляемых и принимаемых данных
-mrtu	выбран протокол Modbus RTU
-pnone	без проверки контроля четности
-s2	стоповых битов 2
/dev/ttyRS485-1	адрес serial-порта, к которому подключено опрашиваемое устройство
-a2	адрес устройства, 2
-t0x03	адрес функции чтения из holding-регистра
-r128	адрес регистра, значение которого мы запрашиваем

Ответ:

```
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[02][03][00][80][00][01][85][D1]
Waiting for a confirmation...
<02><03><02><00><02><7D><85>
SUCCESS: read 1 of elements:
Data: 0x0002
```

Запись нового адреса

Записываем новый адрес устройства в регистр 128, используя функцию 0x06 (Write Single Register)

Записываем новый адрес устройства в регистр 120, используя функцию 0x00 (write single register).

В примере используется широковещательный адрес 0. Использование примера в таком виде *изменит адрес на всех устройствах Wiren Board*, подключенных к порту /dev/ttyRS485-1. Чтобы этого не произошло — отсоедините другие устройства от шины.

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a0 -t0x06 -r128 2
```

Где 0 — широковещательный адрес, а 2 — адрес, который нужно задать.

Ответ:

```
Data to write: 0x2
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[00][06][00][80][00][02][08][32]
Waiting for a confirmation...
ERROR Connection timed out: select
ERROR occured!
```

Сообщение об ошибке возникает всегда, когда запись производится на специальный (широковещательный) адрес 0 (-a0). Теперь к устройству нужно обращаться по адресу 2.

Пример **неправильного** использования команды:

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a0 -t0x06 -r128
```

Здесь не указано значение, которое нужно записать в регистр адреса, поэтому устройство получит неизвестное значение.

Чтение сигнатуры устройства

Прочтем регистры релейного модуля WB-MR14 с адресом 1, содержащие сигнатуру (модель) устройства: WBMR14. Известно, что сигнатура хранится по адресу 200 и занимает 6 регистров.

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x03 -r200 -c 6
```

Ответ:

```
Opening /dev/ttyAPP1 at 9600 bauds (N, 8, 2)
[01][03][00][C8][00][06][44][36]
Waiting for a confirmation...
<01><03><0C><00><57><00><42><00><4D><00><52><00><31><00><34><D4><76>
SUCCESS: read 6 of elements:
Data: 0x0057 0x0042 0x004d 0x0052 0x0031 0x0034
```

В ответе мы получили шесть 16-битных значений, в каждом из которых содержится код одного ASCII-символа. Преобразуем их:

```
echo -e $(modbus_client -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x03 -r200 -c 6 | grep Data | sed -e 's/. *Data:/' -e 's/ 0x00/\\x/g')
```

Ответ:

```
WBMR14
```

Чтение версии прошивки

Прочтем версию прошивки из модуля с modbus-адресом 189. По адресу 250 хранится null-terminated строка максимальной длиной в 16 регистров. Прочтем 16 регистров, начиная с адреса 250, и преобразуем полученный шестнадцатеричный ответ в символьную строку:

```
echo -e $(modbus_client -mrtu -pnone -s2 /dev/ttyRS485-1 -a189 -t0x03 -r250 -c 16 | grep Data | sed -e 's/. *Data:/' -e 's/ 0x00/\\x/g')
```

В результате выполнения команды получаем строку, например **1.3.1**.

Настройка параметров трансформаторов

Для настройки трансформаторов запишите нужные значения в регистры счётчика. Номера регистров смотрите в карте регистров счётчика.

В примере задаются параметры трёх трансформаторов, подключенных к первому каналу счётчика WB-MAP12E(H).

Трансформатор на фазе	Коэффициент трансформации	Фазовый сдвиг
L1	3001	501

L2	3002	502
L3	3003	503

Настройки записываются в память конкретного WB-MAP один раз:

```
$ modbus_client --debug -mrtu -pnone -b9600 -s2 /dev/ttyRS485-2 -a1 -t0x10 -r0x1460 3001 3002 3003 501 502 503
```

Включение реле релейного модуля

На модуле WB-MR14 включим реле с номером 6 (адреса регистров флагов начинаются с нуля, помним об этом!). Используем для этого команду 0x05 (Write Single Coil):

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x05 -r5 1
```

Ответ:

```
Data to write: 0x1
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[01][05][00][05][FF][00][9C][3B]
Waiting for a confirmation...
<01><05><00><05><FF><00><9C><3B>
SUCCESS: written 1 elements!
```

Обратите внимание, утилита modbus_client при записи заменила 1 на 0x00FF, поскольку именно это значение служит для включения реле. Любое ненулевое значение будет заменено на 0x00FF, поэкспериментируйте.

Одновременное включение нескольких реле

Включим все нечетные реле и выключим все четные. Для этого используем функцию 0x0F (Write Multiple Coils). В модуле всего 14 реле, так что мы должны передать значения для 14 регистров с 0 по 13.

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x0F -r0 -c 14 255 0 255 0 255 0 255 0 255 0 255 0 255 0
```

Ответ:

```
Data to write: 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00 0xff 0x00
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[01][0F][00][00][00][0E][02][55][15][1A][97]
Waiting for a confirmation...
<01><0F><00><00><00><0E><02><55><15><1A><97>
SUCCESS: written 14 elements!
```

Обратите внимание на структуру данных запроса:

- [01] — адрес
- [0F] — код функции Write Multiple Coils
- [00][00] — адрес первого регистра флагов для записи
- [00][0E] — количество элементов для записи (14)
- [02] — количество байт данных (14 бит помещаются в 2 байтах)

- [55][15] — 01010101 00010101 (первое реле — младший бит первого байта, 8 реле — старший бит первого байта, 9 реле — младший бит второго байта)
- [1A][97] — CRC16

А так же на структуру ответа:

- <01> — адрес
- <0F> — код функции Write Multiple Coils
- <00><00> — адрес первого регистра флагов для записи
- <00><0E> — количество записанных регистров флагов
- <D4><0F> — CRC16

Подробнее описание структуры данных запросов и ответов можно найти на странице [Протокол Modbus](#).

Настройка взаимодействия входов и выходов реле

Примеры смотрите в статье [Примеры настройки взаимодействия входов и выходов](#).

Работа с Modbus-устройствами Wiren Board без контроллера

Contents

Аппаратная часть

Подготовка к работе

ОС Windows

Настройка порта

Настольный компьютер с Linux

Работа из ОС Windows с помощью Modbus Poll

Настройка соединения

Чтение значений из регистров

Считывание одного регистра

Считывание нескольких регистров подряд

Запись в регистр

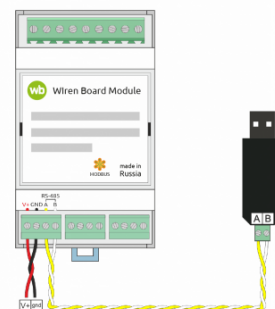


Схема подключения modbus-устройств через адаптер USB-RS485. Если у

Аппаратная часть

Большинство устройств Wiren Board могут работать без управления контроллером — достаточно подать питание на клеммы «V+» и «GND». Но для их настройки и считывания данных потребуется подключиться к ним по протоколу Modbus. Для этого вы можете использовать компьютер с ОС Windows или Linux и адаптер USB-RS485.

Чтобы начать обмен с modbus-устройством, нужно клеммы **A** и **B** устройства подключить ко входам адаптера, подать на устройство питание и настроить программное обеспечение на компьютере.

Подготовка к работе

Независимо от используемой операционной системы вам нужно знать modbus-адрес устройства, коды функций чтения и записи регистров, а также адреса регистров устройства. Перечень общих для всех устройств Wiren Board регистров можно найти в таблице общих регистров. Полный список регистров для каждого устройства смотрите в документации к нему.

ОС Windows

Для подключения по протоколу Modbus из ОС Windows мы рекомендуем использовать утилиту Modbus Poll (<https://www.modbus-tools.com/download.html>) из комплекта Modbus Tools. Она может одновременно опрашивать несколько устройств на шине и отправлять на них данные.

Программа платная, но с бесплатным пробным периодом в 30 дней.

Можно также использовать termite (<http://s2-team.ru/wrkr/prods/modbus-tools/termite/>) — есть «Pro» и «Free» версия.

Настройка порта

Перед подключением к устройству нужно настроить USB-RS485 адаптер: установить драйвер и указать параметры:

- Вставьте адаптер USB-RS485 в USB-порт компьютера.
- Откройте на компьютере **Диспетчер устройств**, для этого кликните правой кнопкой мыши на меню «Пуск» и выберите пункт «Диспетчер устройств».
- Найдите в дереве тип устройств **Порты (COM и LPT)**, разверните ветку и найдите в ней свой адаптер. Если устройство выделено восклицательным знаком — это значит, что драйвер не был установлен автоматически. Установите его вручную по инструкции производителя.
- Если драйвер установлен успешно, то выделите адаптер и в контекстном меню правой кнопки мыши выберите пункт «Свойства».
- В открывшемся окне, на вкладке **Настройки порта** укажите параметры: **Бит в секунду** — 9600, **Биты данных** — 8, **Четность** — Нет, **Стоповые биты** — 2. Если на вкладке имеется «флажок RS485», то включите его.

Настольный компьютер с Linux

Для работы с modbus-устройством используется утилита `modbus_client`.

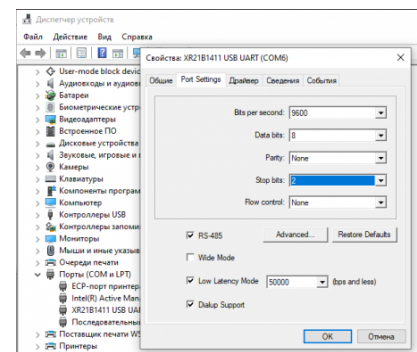
Скачайте пакет для настольных компьютеров с Linux (https://github.com/contactless/modbus-utils/releases/download/1.2/modbus-utils_1.2_amd64.deb).

Перейдите в папку со скаченным пакетом и установите его командой:

```
sudo apt install ./modbus-utils_1.2_amd64.deb
```

Также автоматически должен установиться пакет `libmodbus`, если этого не произошло — установите его из репозитория apt.

Как работать и примеры использования смотрите в статье `modbus_client`.



Настройка порта в диспетчере устройств

Работа из ОС Windows с помощью Modbus Poll

Рассмотрим работу с устройством по протоколу Modbus на примере трехфазного счетчика электроэнергии WB-MAP3E.

Настройка соединения

Перед началом работы нужно настроить соединение: выберите в меню **Connection** — **Connect** и в открывшемся окне укажите

перед началом работы нужно настроить соединение. выберите в меню **Connection** → **Connect** и в открывшемся окне укажите параметры соединения.

При запуске программы у вас уже будет открыто окно опроса устройства. Если вы его закрыли или вам нужно опросить еще одно устройство — выберите в меню **File** → **New**. Окно опроса устройства содержит таблицу, в строках которой выводится пара «Имя регистра» — «Значение».

Чтение значений из регистров

Чтобы вывести значение регистра в таблицу, нужно указать его тип, адрес и другие параметры опроса. Для этого в таблице выберите строку, вызовите контекстное меню правой кнопки мыши и кликните на пункте **Read/Write Definition**. Имя регистра необязательно и нужно для удобства восприятия информации, вводится вручную. Чтобы ввести имя, кликните дважды на ячейке и введите текст с клавиатуры.

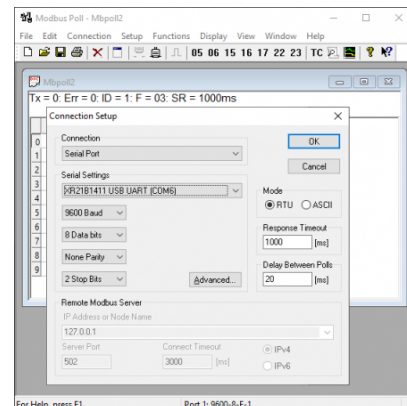
Считывание одного регистра

Для примера считаем из устройства значение одного Holding-регистра. В табличной части окна опроса выберите первую строку, кликните правой кнопкой мыши и в контекстном меню выберите **Read/Write Definition**.

Заполните параметры опроса регистра:

- **Slave id** — modbus адрес устройства, напечатан на наклейке и имеет вид «Addr: XX».
- **Function** — мы хотим считать holding-регрстр, поэтому выберите **03: Read Holding Registers (4x)**. Тип регистра можно узнать из таблицы в документации на устройство.
- **Address mode** — формат адреса регистра. Зависит от того, в каком виде представлен адрес регистра в документации на устройство.
- **Address** — адрес регистра можно взять из таблицы регистров устройства. Мы считаем общий для всех наших устройств регистр — 110. Список общих регистров можно посмотреть на странице Общие Modbus регистры. Обратите внимание на формат адреса — в нашем случае он десятичный. Если формат будет в шестнадцатеричном формате, то измените значение настройки **Address mode**.
- **Quantity** — количество считываемых последовательно регистров, начиная с регистра, указанного в поле **Address**. Мы будем считывать один регистр — установите значение «1».
- **Scan Rate** — период опроса регистра. Оставьте по умолчанию.
- **Address in Cell** — если вы хотите вывести адрес регистра в ячейку рядом со значением — отметьте этот флажок.

Остальные значения оставьте по умолчанию. Нажмите кнопку **OK**.



Настройка соединения в программе Modbus Poll

	Name	Value
0	00110	110 = 96
1		
2		
3		
4		
5		
6		
7		
8		
9		

Считанное значение общего регистра с адресом «100»

Считывание нескольких регистров подряд

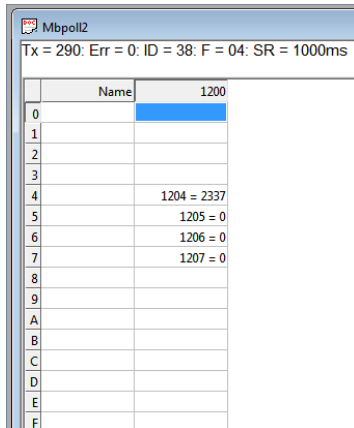
Считаем значение параметра «Прямая активная энергия для фазы L1». Находим его в таблице регистров WM-МАР3Е: тип параметра — Input, разрядность — u64 (занимает 4 регистра), адрес первого регистра — 0x1204.

Заполните параметры опроса регистров:

- **Slave id** — 38.
- **Function** — мы хотим считать input-регистр, поэтому выберите **04: Read Input Registers (3x)**.
- **Address mode** — так как адрес первого регистра в шестнадцатеричном формате, выберите **Hex**.
- **Address** — 1204. Адрес вводится без «0x».
- **Quantity** — значение хранится в четырех регистрах, поэтому установите «4».
- **Address in Cell** — установим флажок, чтобы вывести адреса регистров в ячейки.

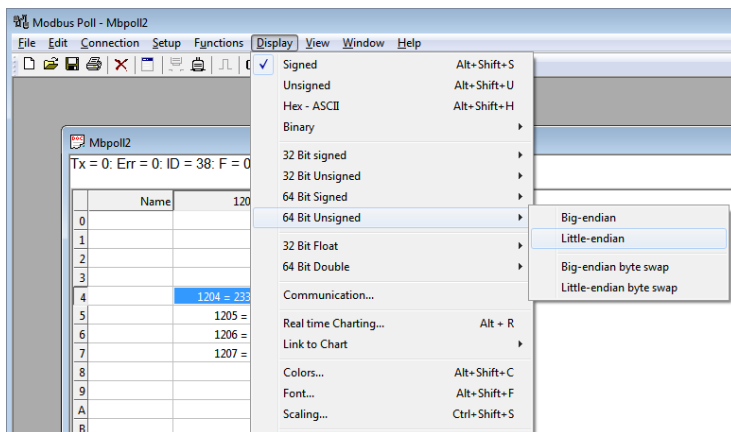
Остальные значения оставьте по умолчанию. Нажимаем кнопку **OK**.

Так как значение хранится в нескольких регистрах, то конечный результат нужно вычислить. Вы можете вычислить значение вручную или автоматически конвертировать. Чтобы конвертировать значение регистров в десятичный вид, перейдите в меню **View** → выберите **64 Bit Unsigned** → **Little-endian**. Теперь десятичное значение будет отображено в первом регистре последовательности.

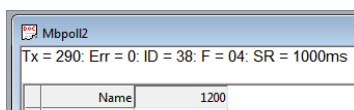


Name	Value
0	1200
1	
2	
3	
4	1204 = 2337
5	1205 = 0
6	1206 = 0
7	1207 = 0
8	
9	
A	
B	
C	
D	
E	
F	

Считанный параметр «Прямая активная энергия для фазы L1» из счетчика WM-MAP3E. Шестнадцатеричный вид.



Переключение отображения параметров в десятичный вид



Name	Value
0	1204 = 2337
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	

0		
1		
2		
3		
4	1204 = 8457	
5	1205 = --	
6	1206 = --	
7	1207 = --	
8		
9		
A		
B		
C		
D		
E		
F		

Считанный параметр «Прямая активная энергия для фазы L1» из счетчика WM-МАРЗЕ. Десятичный вид.

Запись в регистр

Для демонстрации записи в регистр, изменим адрес modbus-устройства.

Откройте окно опроса устройства и вызовите окно записи данных:

- откройте новое окно опроса устройства: меню **File** → **New**;
- выберите в главном меню **Functions** → **Write Single Register**.

В открывшемся окне заполните поля:

- **Slave id** — введите текущий адрес устройства;
- **Address** — введите регистр, где хранится адрес modbus — 128 (десятичный);
- **Value** — введите новый адрес устройства;
- **Use Function** — установите значение **06: Write single register**.

Для отправки данных в устройство нажмите кнопку **Send**.

Запись нового адреса modbus-устройства

Протокол Modbus

- English
- русский

Contents

Основные понятия

Структуры данных Modbus

Модель данных Modbus

Адреса регистров

Нестандартная адресация

Пример описания регистров в документации

Коды функций чтения и записи регистров

Формат данных запросов и ответов Modbus

Коды исключений (ошибки) Modbus

Основные понятия

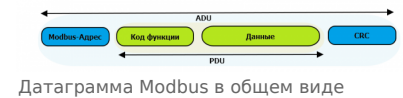
Modbus - это протокол прикладного (седьмого) уровня модели OSI. Чаще всего он служит для обмена данными между устройствами автоматизации и реализован в виде "протокола ответов на запросы (request-reply protocol)".

В устройствах Wipac Board данные Modbus передаются по последовательным линиям связи RS-485. В последовательных линиях связи протокол RS-485 полудуплексный и работает по принципу «клиент-сервер». Каждое устройство в сети (кроме ведущего см. далее) имеет адрес от 1 до 247, адрес 0 используется для широковещательной передачи данных всем устройствам, а адреса 248-255 считаются зарезервированными согласно спецификации Modbus, их использование не рекомендуется.

Существует две спецификации протокола: Modbus RTU и Modbus ASCII. В Modbus RTU передается 11-битный символ, состоящий из 1 стартового бита, 8 бит данных (начиная с младшего бита), бит четности (необязателен) и 2 стоповых бита - если бит четности не передается, или 1 стоповый бит - если бит четности передается. Такой символ передает 1 байт данных. В устройствах Wipac Board по умолчанию бит контроля четности не передается и используется 2 стоповых бита. В Modbus ASCII каждый байт передается двумя символами, представляющими ASCII-коды младшей и старшей четырехбитной группы байта (пример). Modbus RTU передает больше информации при той же скорости последовательной линии, и в устройствах Wipac Board используется именно он. Все дальнейшее описание относится к Modbus RTU.

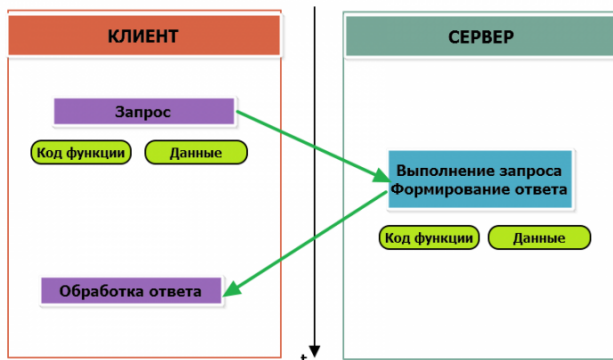
Ведущее устройство ("мастер", или "клиент") периодически опрашивает "ведомое", или "сервер". Ведущее устройство не имеет адреса, передача сообщений от устройства-сервера ведущему без запроса ведущего в протоколе не предусмотрена.

Пакет данных Modbus выглядит, как это показано на рисунке. **PDU** (Protocol Data Unit) — общая часть пакета MODBUS, включающая код функции и данные пакета. **ADU** (Application Data Unit) — полный пакет MODBUS. Включает в себя специфичную для физического уровня часть пакета и PDU. Для последовательных линий в заголовке ADU передается адрес устройства, а в конце — контрольная сумма CRC16. Максимальный размер ADU в последовательных коммуникационных линиях составляет **253 байта** (из максимальных, разрешенных спецификацией 256 байт вычитается 1 байт адреса и два байта контрольной суммы). Для справки — в Modbus TCP максимальная длина пакета составляет 260 байт.



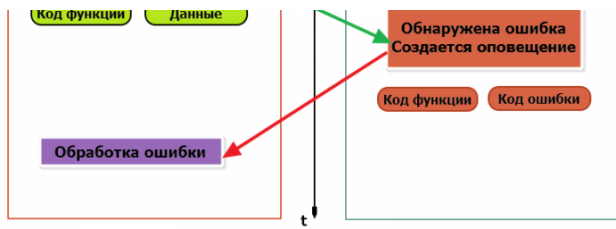
Датаграмма Modbus в общем виде

Функция кодируется одним байтом и определяет, какое действие должно выполнить устройство-сервер. Значение кодов функций лежат в диапазоне от 1 до 255, причем коды от 128 до 255 зарезервированы для сообщений об ошибках со стороны устройства-сервера. Код 0 не используется. Размер блока данных может варьироваться от нуля до максимально допустимого. Если обработка запроса прошла без ошибок, то устройство-сервер возвращает пакет ADU, содержащий запрошенные данные.



Modbus-транзакция, прошедшая без ошибок





Modbus-транзакция с ошибками

При возникновении ошибки устройством возвращается код ошибки. При обычной транзакции код функции в ответе возвращается без изменений; при ошибке старший бит кода функции устанавливается в единицу (то есть *код функции* + 0x80). Так же есть таймаут ожидания ответа от ведомого устройства — бессмысленно долго ждать ответ, который, возможно, никогда и не придет.

Структуры данных Modbus

В Modbus принято кодировать адреса и данные в формате big-endian, то есть в формате, когда байты следуют, начиная со старшего: например, при передаче шестнадцатеричного числа 0x1234 сначала устройством будет принят байт 0x12, а затем — 0x34. Для передачи данных другого типа, например, чисел с плавающей запятой (float), текстовых строк, даты и времени суток и т.п. производитель может выбрать свой собственный способ кодирования — для расшифровки получаемых данных важно ознакомиться со спецификацией производителя устройства.

Модель данных Modbus

Обмен данными с Modbus-устройствами происходит через регистры. В протоколе Modbus определяется четыре типа регистров, показанных в таблице:

Таблица	Размер	Доступ
Регистры флагов (Coils)	1 бит	чтение и запись
Дискретные входы (Discrete Inputs)	1 бит	только чтение
Регистры хранения (Holding Registers)	16-битное слово	чтение и запись
Регистры ввода (Input Registers)	16-битное слово	только чтение

Регистры флагов (Coils) хранят однобитные значения - то есть могут находиться в состоянии 0 или 1. Такие регистры могут обозначать текущее состояние выхода (включено реле). Название "coil" буквально и означает обмотку-актюатор электромеханического реле. Регистры флагов допускают как чтение, так и запись.

Дискретные входы (Discrete Inputs) также являются однобитными регистрами, описывающими состояние входа устройства (например, подано напряжение — 1). Эти регистры поддерживают только чтение.

Регистры хранения (Holding Registers) и **регистры ввода** (Input Registers) представлены двухбайтовым словом и могут хранить значения от 0 до 65535 (0x0000 — 0xFFFF). Регистры ввода допускают только чтение (например, текущее значение температуры). Регистры хранения поддерживают как чтение, так и запись (для хранения настроек). В настоящее время во многих устройствах, в частности в устройствах Wiren Board, эти регистры не разделяются. Команды на чтение регистра хранения N и регистра ввода N обратятся к одному и тому же значению в адресном пространстве устройства.

Адреса регистров

Регистры в стандарте Modbus адресуются с помощью 16-битных адресов. Адресация начинается с нуля. Адрес регистра, таким образом, может принимать значения от 0 до 65535.

Адресные пространства регистров, также называемые таблицами или блоками, могут быть различны для всех четырёх типов регистров. Это значит, что значения регистров с одинаковым адресом, но разным типом, в общем случае разные.

Например, при чтении регистра флагов (coil) номер 42, регистра дискретного входа (Discrete), регистров ввода и хранения (Input и Holding) с теми же адресами, можно получить четыре разных значения.

Нестандартная адресация

В документации на некоторые, особенно старые, устройства адреса элементов (регистров) указываются в формате, не соответствующем стандарту. В этом формате тип элемента кодируется первой цифрой адреса, а адресация начинается не с нуля.

Например, регистр хранения с адресом 0 может записываться как 40001 или 400001, а Coil с адресом 0 как 000001.

В таблице перевода адресов в стандартный формат показаны диапазоны для двух разных нестандартных типов указания адресов и соответствующие им типы данных и диапазоны стандартных адресов.

Тип данных	Стандартные адреса	Стандартные адреса (hex)	Нестандартные адреса (5 цифр)	Нестандартные адреса (6 цифр)
Флаги (Coils)	0-65535	0x0000-0xFFFF	00001-00000	000001-065536

Флагов (Coins)	0-05535	0x0000 - 0xFFFF	00001 - 09999	000001 - 065536
Дискретных входов (Discrete)	0-65535	0x0000 - 0xFFFF	10001 - 19999	100001 - 165536
Регистры входов (Input Registers)	0-65535	0x0000 - 0xFFFF	30001 - 39999	300001 - 365536
Регистры хранения (Holding Registers)	0-65535	0x0000 - 0xFFFF	40001 - 49999	400001 - 465536

Признаки использования нестандартной адресации:

- Адреса записываются в десятичном формате
- Во всех адресах пять или шесть цифр
- Адреса с недискретными данными (показания датчиков и т.п.) начинаются на 30 или 40

Часто рядом с нестандартными адресами указываются и адреса соответствующие стандарту, обычно в шестнадцатеричном формате. Стоит отметить, что физически в пакете данных передаются адреса в стандартном формате, независимо от способа представления их в документации.

Пример описания регистров в документации

В готовых шаблонах устройств для контроллера Wiren Board есть шаблон для однофазного счетчика электроэнергии SDM220 (/usr/share/wb-mqtt-serial/templates/config-sdm220.json). В документации от производителя "Eastron SDM 220 Modbus Smart Meter Modbus Protocol Implementation V1.0" перечислены регистры и соответствующие им измеряемые параметры, например:

Address (Register)	Description	Units	Modbus Protocol Start Address Hex (Hi Byte Lo Byte)
30001	Line to neutral volts.	Volts	00 00
30007	Current.	Amps.	00 06
30013	Active power	Whatts	00 0C
30019	Apparent power	VoltAmps	00 12
...

Производитель в таблице приводит и логические, и физические адреса регистров, что позволяет нам с легкостью создать шаблон устройства и проиллюстрировать связь между логическими и физическими адресами Modbus-регистров.

```

"channels" : [
  {
    "name" : "Voltage",
    "type" : "voltage",
    "reg_type" : "input",
    "address" : "0x00",
    "format" : "float"
  },
  {
    "name" : "Current",
    "type" : "current",
    "reg_type" : "input",
    "address" : "0x06",
    "format" : "float"
  },
  {
    "name" : "Active Power",
    "type" : "power",
    "reg_type" : "input",
    "address" : "0x0c",
    "format" : "float"
  },
  {
    "name" : "Apparent Power",
    "type" : "power",
    "reg_type" : "input",
    "address" : "0x12",
    "format" : "float"
  }
],

```

Фрагмент шаблона счетчика SDM220

Коды функций чтения и записи регистров

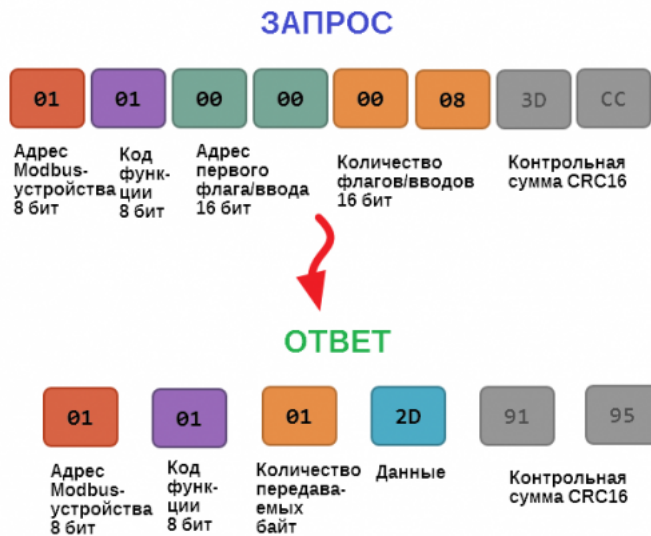
В следующей таблице приведены наиболее распространенные коды функций Modbus:

Код функции	HEX	Название	Действие
1	0x01	Read Coils	Чтение значений нескольких регистров флагов
2	0x02	Read Discrete Inputs	Чтение значений нескольких дискретных входов
3	0x03	Read Holding Registers	Чтение значений нескольких регистров хранения
4	0x04	Read Input Registers	Чтение значений нескольких регистров ввода
5	0x05	Write Single Coil	Запись одного регистра флагов
6	0x06	Write Single Register	Запись одного регистра хранения
15	0x0F	Write Multiple Coils	Запись нескольких регистров флагов
16	0x10	Write Multiple Register	Запись нескольких регистров хранения

Команды условно можно разделить по типам: чтение значений — запись значений; операция с одним значением — операция с несколькими значениями.

Формат данных запросов и ответов Modbus

Рассмотрим подробнее, как происходит обмен данными между устройством-клиентом, отправляющим запрос, и устройством-сервером, отвечающим ему. На следующем рисунке показан обмен данными контроллера с устройством с адресом 0x01. Мы хотим прочесть 8 coil-регистров, начиная с первого.



Обмен данными в Modbus

В качестве данных мы получили шестнадцатеричное число 0x2D, то есть состояние восьми coil-регистров в двоичном виде такое: 0b10110100.

В следующей таблице приведены структуры данных запросов и ответов для основных функций Modbus.

Код функции	Запрос	Ответ
1 (Read Coils) и 2 (Read Discrete Inputs)	<ul style="list-style-type: none"> Адрес первого регистра флагов или входного регистра (16 бит) Количество данных (8 значений на байт) (16 бит) 	<ul style="list-style-type: none"> Число передаваемых байт (8 бит) Значения регистров флагов или входных регистров (8 значений на байт)
3 (Read Holding Registers) и 4 (Read Input Registers)	<ul style="list-style-type: none"> Адрес первого регистра (16 бит) Количество регистров, которые нужно прочесть 	<ul style="list-style-type: none"> Число передаваемых байт (8 бит) Значения регистров (16 бит на 1 регистр)
5 (Write Single Coil)	<ul style="list-style-type: none"> Адрес регистра (16 бит) Значение, которое нужно записать (0 — выключить, 0xFF00 — включить) 	Ответ аналогичен запросу
6 (WriteSingle Register)	<ul style="list-style-type: none"> Адрес регистра(16 бит) Новое значение регистра (16 бит) 	Ответ аналогичен запросу
	<ul style="list-style-type: none"> Адрес первого регистра флагов для записи (16 бит) 	

15 (WriteMultipleCoils)	<ul style="list-style-type: none"> Количество регистров флагов для записи (16 бит) Количество передаваемых байт данных для регистров флагов (8 бит) Данные (8 регистров флагов на байт) 	<ul style="list-style-type: none"> Адрес первого coil-регистра (16 бит) Количество записанных coil-регистров(16 бит)
16 (Write Multiple register)	<ul style="list-style-type: none"> Адрес первого регистра хранения для записи (16 бит) Количество регистров хранения для записи (16 бит) Количество передаваемых байт данных для регистров (8 бит) Данные (16 байт на регистр) 	<ul style="list-style-type: none"> Адрес первого регистра хранения (16 бит) Количество записанных регистров хранения(16 бит)

Коды исключений (ошибки) Modbus

Если запрос не может по той или иной причине быть обработан устройством-сервером, то в ответ он отправляет сообщение об ошибке. Сообщение об ошибке содержит адрес Modbus-устройства, код функции, при выполнении которой произошла ошибка, увеличенный на 0x80, код ошибки и контрольную сумму:

ОШИБОЧНЫЙ ЗАПРОС



СООБЩЕНИЕ ОБ ОШИБКЕ



Транзакция завершилась с ошибкой

В этом случае мы попытались обратиться к несуществующему адресу регистра 0xFFFF и попытались прочесть 8 регистров флагов. В результате мы получили код ошибки 0x03 — "В поле данных передано неверное значение".

Наиболее распространенные коды ошибок Modbus приведены в следующей таблице:

Код ошибки	Название ошибки	Что означает
1	Illegal Function	В запросе был передан недопустимый код функции
2	Illegal Data Address	Указанный в запросе адрес не существует
3	Illegal Data Value	Неверный формат запроса, например количество байт в запросе не соответствует ожидаемому. Примечание: несмотря на название, эта ошибка не говорит о том, что само значение регистра неправильное или ошибочное, и должна использоваться только для ошибок формата запроса.
4	Server Device Failure	Произошла невосстановимая ошибка на устройстве при выполнении запрошенной операции
5	Acknowledge	Запрос принят, выполняется, но выполнение потребует много времени; необходимо увеличить таймаут.
6	Server Device Busy	Устройство занято обработкой предыдущего запроса.
7	Negative Acknowledge	Устройство не может выполнить запрос, необходимо получить от устройства дополнительную диагностическую информацию. Возможно, требуется тех. обслуживание.
8	Memory Parity Error	Ошибка четности при обращении к внутренней памяти устройства.

Вычисление контрольной суммы Modbus

Для протокола Modbus RTU 16-битная контрольная сумма (CRC) вычисляется по алгоритму, описанному в спецификации Modbus, в документе "Modbus Serial Line Protocol and Implementation Guide", раздел "CRC-generation". Передающее устройство формирует два байта контрольной суммы на основе данных сообщения, а принимающее устройство заново вычисляет контрольную сумму и сравнивает с полученной. Совпадение принятой и вычисленной контрольной суммы Modbus RTU считается индикатором успешного обмена данными.

В случае ограниченных вычислительных ресурсов для вычисления контрольной суммы существует функция, использующая табличные значения (также приведена в спецификации).

Настройка параметров подключения по RS-485 для Modbus-устройств Wiren Board

- English
- русский

Contents

Введение

Параметры порта по умолчанию

Изменение скорости обмена

Смена уровня доступа к веб-интерфейсу

Настройка

Настройка параметров обмена

Если параметры подключения неизвестны

Введение

Устройства Wiren Board управляются по протоколу Modbus RTU и на физическом уровне подключаются через интерфейс RS-485.

Параметры порта по умолчанию

Значение по умолчанию	Название параметра в веб-интерфейсе	Параметр
9600	Baud rate	Скорость, бит/с
8	Data bits	Количество битов данных
None	Parity	Бит чётности
2	Stop bits	Количество стоповых битов

Изменение скорости обмена

Скоро в стабильном релизе, а пока доступно в testing

Для ускорения отклика устройств на шине RS485 рекомендуем поднять скорость обмена до 115 200 бит/с.

Отметим, что низкая скорость обмена прощает многие ошибки построения шины, но на высоких скоростях выполнение рекомендаций по построению шины обязательно.

Смена уровня доступа к веб-интерфейсу

Для изменения настроек контроллера у вас должен быть уровень доступа *Administrator*.

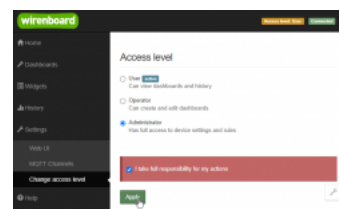
Изменить его можно в разделе **Settings** → **Change access level**.

После завершения настроек рекомендуем поставить уровень доступа *User* или *Operator* — это поможет не совершить случайных ошибок при ежедневной работе с веб-интерфейсом.

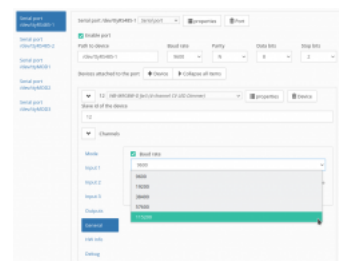
Настройка

Увеличим скорость обмена в Modbus-устройствах Wiren Board со значения по умолчанию до 115 200 бит/с:

1. Подключите и настройте все устройства на скорости 9600 бит/с, которая стоит у них по умолчанию.
2. Убедитесь, что все работает как надо: данные идут со всех устройств, каналы не горят красным, в системном журнале нет ошибок порта.
3. Откройте веб-интерфейс контроллера и перейдите **Settings** → **Configs** → **Serial Device Driver Configuration**.
4. Выберите нужный порт, в параметрах устройства в группе **General** поставьте флажок **Baud rate** и выберите желаемую скорость обмена: 115 200 бит/с. Скорость порта пока оставьте прежней.
5. Вверху страницы нажмите на кнопку **Save**, это запишет новое значение скорости в устройство. Но так как порт работает на старой скорости, то устройства отвечать не будут.



Уровень «Администратор»



Выбор желаемой скорости обмена в настройках устройства

6. Укажите в настройках порта ту же скорость, которую вы выбрали в настройках устройства: 115 200 бит/с.

7. Снова сохраните настройки. Теперь настройки устройства и порта совпадают, устройство должно начать отвечать.

Настройка параметров обмена

Чтобы изменить параметры подключения, нам понадобится:

- знать текущие настройки подключения устройства;
- контроллер с утилитой `modbus_client` или компьютер с адаптером USB-RS485 и программой для работы с Modbus;
- номера регистров, которые описаны в таблице общих регистров.

Подготовка:

1. Подключите устройство по шине RS-485 к контроллеру или другому оборудованию, где будете выполнять команды.
2. Если вы выполняете команды на контроллере:

- откройте консоль контроллера по SSH,
- остановите драйвер `wb-mqtt-serial`.

3. Можно менять настройки устройств.

Допустим, у нас есть Modbus-устройство Wiren Board с заводскими параметрами подключения, Modbus-адресом 1 и подключённое к порту `/dev/ttyRS485-1`.

Изменим адрес устройства, для этого запишем в регистр 128 новый адрес, например 12:

```
modbus_client --debug -mrtu -b9600 -pnone -s2 /dev/ttyRS485-1 -a1 -t0x06 -r128 12
```

Теперь изменим скорость порта устройства с 9600 бит/с на 115 200 бит/с, для этого запишем в регистр 110 новое значение, формат которого можно посмотреть в таблице общих регистров:

```
modbus_client --debug -mrtu -b9600 -pnone -s2 /dev/ttyRS485-1 -a1 -t0x06 -r110 1152
```

Теперь устройство передаёт и принимает данные на скорости 115 200 бит/с.

Остальные параметры меняются аналогично: смотрите, в каком регистре хранится значение и записываете в него новое.

Если параметры подключения неизвестны

Бывает так, что параметры подключения устройства неизвестны, то можно или сбросить их к заводским, или узнать перебором, для этого загрузите на контроллер скрипт `Perebor.sh.tar.gz` и выполните его. Если адрес, к которому подключено устройство отличается от `/dev/ttyRS485-1`, измените его в теле скрипта.

Как это работает: мы обращаемся к регистру 128, в котором во всех modbus-устройствах Wiren Board хранится modbus-адрес. Вывод скрипта будет содержать строки, подобные этим:

```
Speed:9600      Stop bits:1    Parity:none    Modbus address:0x0001
Speed:9600      Stop bits:2    Parity:none    Modbus address:0x0001
```

Для стоп-битов, скорее всего, вы получите два значения: 1 и 2. Уточнить настройку можно считав значение из регистра 112 с уже известным адресом, скоростью, четностью:

```
modbus_client --debug -mrtu -b9600 -pnone -s2 /dev/ttyAPP1 -a0x01 -t0x03 -r112
```

или

```
modbus_client --debug -mrtu -b9600 -pnone -s1 /dev/ttyAPP1 -a0x01 -t0x03 -r112
```

```
SUCCESS: read 1 of elements:
Data: 0x0002
```

Если при чтении из регистра 112 вы получаете ошибку — устройство не поддерживает изменение параметров подключения. В этом случае для подключения используется значение по умолчанию, 2 стоп-бита.

Modbus-адрес устройства Wiren Board

Contents

Общая информация

Определение адресов всех устройств на шине

Изменение адреса устройству с известным адресом

Изменение адреса устройству с неизвестным адресом

Восстановление доступа

Устройство питается от блока питания

Устройство питается от Vout контроллера

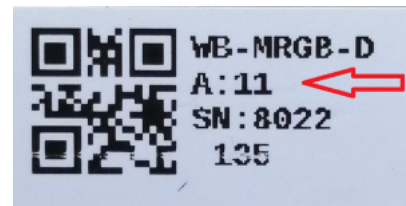
Полезные ссылки

Общая информация

Заводской Modbus-адрес устройства Wiren Board можно узнать на наклейке, которая находится на корпусе устройства.

Если заводской адрес был изменен, то можно воспользоваться одним из способов ниже, для работы вам понадобится утилита `Modbus_client`, которая доступна для контроллеров Wiren Board и компьютеров с ОС Linux. Если у вас компьютер с ОС Windows, то вы можете восстановить доступ к устройству.

ВНИМАНИЕ: если вы выполняете команды на контроллере, то перед началом работы остановите драйвер `wb-mqtt-serial`, а после окончания — запустите снова.



Modbus-адрес, установленный на производстве

Определение адресов всех устройств на шине

Если перебрать все доступные адреса и прочитать регистр с сигнатурой устройства — можно получить список устройств на шине:

1. Подключите устройства по шине RS-485 к контроллеру или другому оборудованию, где будете выполнять команды.
2. Если вы выполняете команды на контроллере:
 - откройте консоль контроллера по SSH,
 - остановите драйвер `wb-mqtt-serial`.
3. Замените в скрипте порт `/dev/ttyRS485-1` на тот, к которому подключены устройства, настройки соединения **9600N2** задаются параметрами `-b9600 -pnone -s2`:

```
for i in {1..247}; do echo -n "$i - "; D=`modbus_client -mrtu /dev/ttyRS485-1 --debug -b9600 -pnone -s2 -a$i -t3 -o100 -r200 -c6 2>/dev/null | grep Data: | awk 'gsub("Data:", "")' | sed -e 's/0x00/\\x/g' -e 's/\\s//g'; echo -e $D; done
```

4. Скопируйте и вставьте измененный скрипт в консоль контроллера, нажмите **Enter**.

Скрипт переберет все адреса с 1 по 247 и выведет в консоль результат для каждого адреса:

```
# for i in {1..247}; do echo -n "$i - "; D=`modbus_client -mrtu /dev/ttyRS485-1 --debug -b9600 -pnone -s2 -a$i -t3 -o100 -r200 -c6 2>/dev/null | grep Data: | awk 'gsub("Data:", "")' | sed -e 's/0x00/\\x/g' -e 's/\\s//g'; echo -e $D; done
1 -
2 -
3 -
4 -
5 -
6 - WBMWAC
7 -
8 -
9 - WBMRGB
10 -
11 -
12 -
...
```

Изменение адреса устройству с известным адресом

Вы можете записать новый адрес в регистр 128 (0x80):

1. Подключите устройство по шине RS-485 к контроллеру или другому оборудованию, где будете выполнять команды.
2. Если вы выполняете команды на контроллере:
 - откройте консоль контроллера по SSH,
 - остановите драйвер `wb-mqtt-serial`.

3. Чтобы назначить новый адрес 12 устройству с адресом 1 и подключенное к порту /dev/ttyRS485-1 выполните команду:

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x06 -r128 12
```

Пример успешного выполнения команды:

```
~# modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a1 -t0x06 -r128 12
Data to write: 0xc
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[01][06][00][80][00][0C][88][27]
Waiting for a confirmation...
<01><06><00><80><00><0C><88><27>
SUCCESS: written 1 elements!
```

Изменение адреса устройству с неизвестным адресом

Если вам достаточно изменить адрес устройства, то вы можете сделать это отправив ему широковещательный запрос.

ВНИМАНИЕ: новый адрес будет установлен для всех устройств на шине, поэтому отключите те устройства, адреса которых вы не хотите менять.

Чтобы изменить адрес, выполните шаги:

1. Подключите устройство по шине RS-485 к контроллеру или другому оборудованию, где будете выполнять команды.
2. Если вы выполняете команды на контроллере:
 - откройте консоль контроллера по SSH,
 - остановите драйвер wb-mqtt-serial.
3. Замените в команде порт /dev/ttyRS485-1 на тот, к которому подключены устройства и выполните команду на контроллере:

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a0 -t0x06 -r128 1
```

Так как команда отправляет данные по широковещательному адресу — сообщение об ошибке в ответе является нормой.

Запишем всем устройствам на шине в регистр 128 (0x80) новый адрес 1:

```
modbus_client --debug -mrtu -pnone -s2 /dev/ttyRS485-1 -a0 -t0x06 -r128 1
Data to write: 0x1
Opening /dev/ttyRS485-1 at 9600 bauds (N, 8, 2)
[00][06][00][80][00][01][48][33]
Waiting for a confirmation...
ERROR Connection timed out: select
ERROR occurred!
```

Восстановление доступа

Вы можете сбросить настройки приемопередатчика Modbus-устройства до заводских: скорость — 9600, чётность (parity) — N, количество стоп-бит — 2, Modbus-адрес — 1.

Это может быть полезно, если вам неизвестны все параметры подключения. Для сброса настроек используется утилита wb-mcu-fw-flasher, которая доступна для контроллеров Wiren Board, а также компьютеров с ОС Linux и Windows.

Устройство питается от блока питания

1. Подключите **только одно устройство** по шине RS-485 к контроллеру или другому оборудованию, где будете выполнять команды.
2. Если вы выполняете команды на контроллере:
 - откройте консоль контроллера по SSH,
 - остановите драйвер wb-mqtt-serial.
3. Отключите питание устройства.
4. Подайте питание на устройство и в течение двух секунд, пока устройство находится в режиме загрузчика, выполните команду, где /dev/ttyRS485-1 (COM1) — порт, к которому подключено устройство:

- на контроллере или компьютере с ОС Linux:

```
wb-mcu-fw-flasher -d /dev/ttyRS485-1 -a0 -u
```

- на компьютере с ОС Windows перейдите в папку с утилитой, а потом выполните команду:



Индикация режима загрузчика

```
wb-mcu-fw-flasher_1.0.3.exe -d COM1 -a0 -u
```

5. Прошейте устройство новой прошивкой, или перезапустите, для этого отключите и включите питание устройства.

Пример успешного сброса настроек приемопередатчика:

```
root@wirenboard-A4DTZKTБ:~# wb-mcu-fw-flasher -d /dev/ttyRS485-1 -a0 -u
/dev/ttyRS485-1 opened successfully.
Send reset UART settings and modbus address command...
Ok.
Device is in Bootloader now! To flash FW run
wb-mcu-fw-flasher -d <port> -f <firmware.wbfw>
```

Устройство питается от Vout контроллера

Если устройство питается от выхода *Vout* контроллера, то вы можете управлять его питанием программно. Этот способ доступен только для контроллеров Wiren Board.

1. Подключите **только одно устройство** по шине RS-485 к контроллеру.
2. Откройте консоль контроллера по SSH.
3. Остановите драйвер `wb-mqtt-serial`.
4. Выполните команду, которая перезагрузит устройство, подключенное к порту `/dev/ttyRS485-1` и сбросит настройки приемопередатчика:

```
mosquitto_pub -t '/devices/wb-gpio/controls/V_OUT/on' -r -m 0 && sleep 3 && mosquitto_pub -t '/devices/wb-gpio/controls/V_OUT/on' -r -m 1 && sleep 1 && wb-mcu-fw-flasher -d /dev/ttyRS485-1 -a0 -u
```

Пример успешного сброса настроек приемопередатчика:

```
~# mosquitto_pub -t '/devices/wb-gpio/controls/V_OUT/on' -r -m 0 && sleep 3 && mosquitto_pub -t '/devices/wb-gpio/controls/V_OUT/on' -r -m 1 && sleep 1 && wb-mcu-fw-flasher -d /dev/ttyRS485-1 -a0 -u
/dev/ttyRS485-1 opened successfully.
Send reset UART settings and modbus address command...
Ok.
Device is in Bootloader now! To flash FW run
wb-mcu-fw-flasher -d <port> -f <firmware.wbfw>
```

Полезные ссылки

- Настройка параметров обмена данными по RS-485 для modbus-устройств Wiren Board
- Обновление прошивки Modbus-устройств Wiren Board
- Описание утилиты `modbus_client`
- Сервисная утилита `wb-mcu-fw-flasher`
- Описание драйвера `wb-mqtt-serial`

Карта регистров модулей реле

Описание

Общие регистры модулей реле

Адрес		Параметры регистра			Канал	Описание	Значения	FW
Dec	Hex	Тип	Доступ	Формат				
0	0x0000	Coil	RW	bool	Канал 1	Состояние канала реле	0 - разомкнут, 1 - замкнут	
1	0x0001				Канал 2			
2	0x0002				Канал 3			
3	0x0003				Канал 4			
4	0x0004				Канал 5			
5	0x0005				Канал 6			
0	0x0000	Discrete input	RO	bool	Канал 1	Состояние входа реле	0 - разомкнут, 1 - замкнут, Error: 0	
1	0x0001				Канал 2			
2	0x0002				Канал 3			
3	0x0003				Канал 4			
4	0x0004				Канал 5			
5	0x0005				Канал 6			
7	0x0007				Канал 0			
4	0x0004	Input	RW	u16		Статус внешнего источника питания (только для реле со встроенным блоком питания)	0: внешний источник питания подключен, 1: нет подключения внешнего источника питания	1.18.0
5	0x0005	Holding	RW	u16		Служебный регистр, значение должно быть 0	0	
6	0x0006	Holding	RW	u16		Режим работы реле при отключении питания	0: не восстанавливать состояние реле, 1: восстанавливать состояние реле, 2: установить состояние реле согласно состоянию входа (только если настроен режим входа 1: выключатель с фиксацией) (с прошивки версии после 1.18.0)	1.5.3
8	0x0008	Holding	RW	u16		Таймаут для безопасного режима	с 0	
9	0x0009	Holding	RW	u16	Вход 1	Режим взаимодействия отдельного цифрового входа с соответствующим релейным выходом. В регистре 5 должно стоять значение по умолчанию (0).	0: кнопки без фиксации, 1: выключатель с фиксацией, 2: отключать все реле при нажатии, 3: отключить взаимодействие, 4: управлять по mapping-матрице, 5: не используется в реле 6: управлять по mapping-матрицам для кнопок <i>До прошивки 1.12.0 режимом по умолчанию был 0.</i>	
10	0x000A				Вход 2			
11	0x000B				Вход 3			
12	0x000C				Вход 4			
13	0x000D				Вход 5			
14	0x000E				Вход 6			

16	0x0010				Вход 0	Режим работы цифрового входа 0 для отключения всех реле	2: отключать все реле при нажатии, 3: отключить взаимодействие, 4: управлять по mapping-матрице	1.9.0
20	0x0014	Holding	RW	u16	Вход 1	Время защиты входа от дребезга	мс 0 - 100 (50) 0 - 250 (50) начиная с 1.17.8	1.13.0
21	0x0015				Вход 2			
22	0x0016				Вход 3			
23	0x0017				Вход 4			
24	0x0018				Вход 5			
25	0x0019				Вход 6			
27	0x001B				Вход 0			
32	0x0020	Input	RO	u16	Вход 1	Счетчик срабатываний входа	0 – 65535 по кругу Error: 0	
33	0x0021				Вход 2			
34	0x0022				Вход 3			
35	0x0023				Вход 4			
36	0x0024				Вход 5			
37	0x0025				Вход 6			
39	0x0027				Вход 0			
40	0x0020	Input	RO	u32	Вход 1	Целая часть значения частоты сигнала	x1.52588*10 ⁻⁰⁵ , Гц Error: 0	1.15.0
42	0x0028				Вход 2			
44	0x002A				Вход 3			
46	0x002C				Вход 4			
48	0x002E				Вход 5			
50	0x0030				Вход 6			
54	0x0032				Вход 0			
41	0x0029	Input	RO	u16	Вход 1	Дробная часть значения частоты сигнала 1 Гц = 65536 (2 ^ 16)	Error: 0	1.17.0
43	0x002B				Вход 2			
45	0x002D				Вход 3			
47	0x002F				Вход 4			
49	0x0031				Вход 5			
51	0x0033				Вход 6			
55	0x0037				Вход 0			
464	0x01D0	Input	RO	u16	Вход 1	Счётчик коротких нажатий	0 – 65535 по кругу Error: 0	
465	0x01D1				Вход 2			
466	0x01D2				Вход 3			
467	0x01D3				Вход 4			
468	0x01D4				Вход 5			
469	0x01D5				Вход 6			
471	0x01D7				Вход 0			
480	0x01E0	Input	RO	u16	Вход 1	Счётчик длинных нажатий	0 – 65535 по кругу Error: 0	
481	0x01E1				Вход 2			
482	0x01E2				Вход 3			
483	0x01E3				Вход 4			
484	0x01E4				Вход 5			
485	0x01E5				Вход 6			
487	0x01E7				Вход 0			
496	0x01F0	Input	RO	u16	Вход 1	Счётчик двойных нажатий	0 – 65535 по кругу Error: 0	
497	0x01F1				Вход 2			
498	0x01F2				Вход 3			
499	0x01F3				Вход 4			
500	0x01F4				Вход 5			
501	0x01F5				Вход 6			
503	0x01F7				Вход 0			
512	0x0200	Input	RO	u16	Вход 1	Счётчик короткого, а затем длинного нажатий	0 – 65535 по кругу Error: 0	
513	0x0201				Вход 2			

514	0x0202				Вход 3			
515	0x0203				Вход 4			
516	0x0204				Вход 5			
517	0x0205				Вход 6			
519	0x0207				Вход 0			
123	0x007B	Input	RO	u16		Напряжение на микроконтроллере	mB Error: 0	1.16.0
124	0x007C	Input	RO	u16		Внутренняя температура микроконтроллера	x0.1, °C Error: 0	1.16.0
384 - 447	0x0180 - 0x01BF	Holding	RW	u16		Регистры mapping-матрицы	0	1.9.0
544 - 607	0x0220 - 0x025F	Holding	RW	u16		Регистры mapping-матрицы для коротких нажатий	0	1.17.0
608 - 671	0x0260 - 0x029F	Holding	RW	u16		Регистры mapping-матрицы для длинных нажатий	0	
672 - 735	0x02A0 - 0x02DF	Holding	RW	u16		Регистры mapping-матрицы для двойных нажатий	0	
736 - 799	0x02E0 - 0x031F	Holding	RW	u16		Регистры mapping-матрицы для сначала короткого, потом длинного нажатий	0	
930	0x03A2	Holding	RW	u16	Канал 1	Настройка аварийного режима при пропадании связи с контроллером. Для всех сигнатур, кроме mr3, mr6, mr6c, mrw3	0 - управлять каналом со входа всегда, 1 - управлять каналом со входа только при аварии, 2 - отключить канал при аварии, 3 - включить канал при аварии.	1.18.2
931	0x03A3				Канал 2			
932	0x03A4				Канал 3			
933	0x03A5				Канал 4			
934	0x03A6				Канал 5			
935	0x03A7				Канал 6			
938	0x03AA	Holding	RW	u16	Канал 1	Настройка аварийного режима при отключении внешнего источника питания. Только для WB-MR6C v.3		1.18.0
939	0x03AB				Канал 2			
940	0x03AC				Канал 3			
941	0x03AD				Канал 4			
942	0x03AE				Канал 5			
943	0x03AF				Канал 6			
1100	0x044C	Holding	RW	u16	Вход 1	Время удержания входа в замкнутом состоянии для фиксации долгого нажатия	мс 500 - 5000 (2000)	1.17.0
1101	0x044D				Вход 2			
1102	0x044E				Вход 3			
1103	0x044F				Вход 4			
1104	0x0450				Вход 5			
1105	0x0451				Вход 6			
1107	0x0453				Вход 0			
1140	0x0474	Holding	RW	u16	Вход 1	Время двойного нажатия	мс 100 - 2000 (500)	
1141	0x0475				Вход 2			
1142	0x0476				Вход 3			
1143	0x0477				Вход 4			
1144	0x0478				Вход 5			
1145	0x0479				Вход 6			
1147	0x047B				Вход 0			

Общие для всех Modbus-устройств Wiren Board регистры

Адрес		Параметры регистра			Описание	Значения
Dec	Hex	Тип	Доступ	Формат		
104-105	0x0068 - 0x0069	Input	RO	u32	Время работы с момента загрузки	секунды
110	0x006E	Holding	RW	u16	Скорость порта RS-485. Настройка параметров подключения по RS-485	x100, Боды 12 — 1200 бит/с, 24 — 2400 бит/с, 48 — 4800 бит/с, 96 — 9600 бит/с , 192 — 19 200 бит/с, 384 — 38 400 бит/с, 576 — 57 600 бит/с, 1152 — 115 200 Кбит/с
111	0x006F	Holding	RW	u16	Настройка бита чётности порта RS-485	0 — нет бита чётности (none) , 1 — нечётный (odd), 2 — чётный (even)
112	0x0070	Holding	RW	u16	Количество стоп-битов порта RS-485	1, 2
120	0x0078	Holding	RW	u16	Сохранение состояния при перезагрузке устройства	0 - сохраняет , >0 - без сохранения
121	0x0079	Input	RO	u16	Текущее напряжение питания	mВ
128	0x0080	Holding	RW	u16	Modbus-адрес устройства (подробнее)	
129	0x0081	Holding	RW	u16	Перевод в режим обновления прошивки на 2 минуты	0 - выключен , >0 - включен
200-205	0x00C8 - 0x00CD	Input	RO	string	Модель устройства	
220-241	0x00DC - 0x00F1	Input	RO	string	Время и дата сборки прошивки	
220-248	0x00DC - 0x00F8	Input	RO	string	Хэш коммита и название ветки откуда собрана прошивка (2 символа в регистре)	
250-265	0x00FA - 0x0109	Input	RO	string	Версия прошивки	
266-269	0x010A - 0x010D	Input	RO	u64	Расширение серийного номера	
270-271	0x010E - 0x010F	Input	RO	u32	Серийный номер	
290-301	0x0122 - 0x012D	Holding	RO	string	Сигнатура прошивки	
330-336	0x014A - 0x0150	Holding	RO	string	Версия загрузчика	

Регистры настройки параметров обмена данными по RS-485 поддерживаются начиная с прошивки версии 1.6.0

Обновление прошивки Modbus-устройств Wiren Board

Contents

Общая информация

Автоматическое обновление

- Обновление всех устройств на шине
- Обновление определенного устройства

Ручное обновление

- Подготовка устройства
- Загрузка прошивки в устройство

Восстановление прошивки устройства

- Автоматически
- Вручную

Полезные ссылки

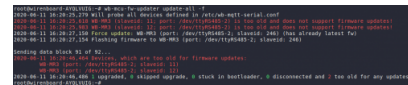
Общая информация

В наших modbus-устройствах реализован механизм загрузчика прошивок — bootloader. Он позволяет обновлять микропрограммы устройств и модулей Wiren Board по RS-485/Modbus RTU.

В режиме загрузчика основные функции устройства отключаются, а коммуникационные параметры в режиме загрузчика фиксированы и не зависят от значений в памяти устройства: 9600 8N2.

Автоматическое обновление

При обновлении прошивки удаляются ИК-команды, сохранённые в устройствах WB-MSW и WB-MIR. Рекомендуем сохранить банки команд перед обновлением с помощью скрипта.



Пример работы wb-fw-mcu-updater

Автоматическое обновление прошивки выполняется с помощью предустановленной на контроллеры Wiren Board утилиты wb-mcu-fw-updater и позволяет установить свежую версию ПО сразу на все подключенные устройства или отдельно на каждое. Определение сигнатуры (модели) устройства, новой прошивки произойдет автоматически.

Для использования утилиты нужен доступ в интернет, если это не так — смотрите раздел про ручное обновление.

Вы можете использовать утилиту и без нашего контроллера, для этого вам понадобится Debian-подобная ОС Linux. Читайте инструкцию по установке в описании утилиты.

Обновление всех устройств на шине

Вы можете обновить все устройства, настроенные в разделе **Serial Devices Configuration** веб-интерфейса (файл /etc/wb-mqtt-serial.conf)

1. Подключите устройства по шине RS-485 к контроллеру.
2. Настройте подключенные устройства в веб-интерфейсе.
3. Откройте консоль контроллера по SSH.
4. Обновите все настроенные устройства командой:

```
wb-mcu-fw-updater update-all
```

Обновление определенного устройства

Чтобы обновить определенное устройство:

1. Подключите устройство по шине RS-485 к контроллеру или компьютеру с ОС Linux.
2. Узнайте modbus-адрес устройства, которое хотите обновить.
3. Откройте консоль контроллера или компьютера с ОС Linux по SSH
4. Запустите утилиту wb-mcu-fw-updater параметрами: ключ update-fw, а также порт и modbus-адрес.

Например, обновим прошивку устройства с modbus-адресом 70 и подключенного к порту /dev/ttyRS485-1:

```
wb-mcu-fw-updater update-fw /dev/ttyRS485-1 -a70
```

Полный список параметров и примеры работы смотрите на странице утилиты.

Ручное обновление

Мы не рекомендуем этот способ, так как выбранная вами версия прошивки может неправильно работать с той версией `wb-mqtt-serial`, которая у вас установлена. Но если на объекте нет доступа в интернет, или у вас устройство с ОС Windows, это единственный вариант.

Ручное обновление можно сделать утилитой `wb-mcu-fw-flasher`, которую нужно предварительно установить. Способ установки отличается и зависит от используемой операционной системы.

ВНИМАНИЕ: если вы выполняете команды на контроллере, то перед началом работы остановите драйвер `wb-mqtt-serial`, а после окончания запустите снова.

Подготовка устройства

Прошивать устройства можно:

- по modbus-адресу устройства.
- по широковещательному адресу — 0.

Для прошивки нескольких устройств на шине нужно поочередно перевести их в режим загрузчика и прошить.

Загрузка прошивки в устройство

Для загрузки прошивки выполните шаги:

1. Подключите устройство по шине RS-485 к контроллеру или другому оборудованию, где установлена утилита прошивки.
2. Если вы выполняете команды на контроллере:
 - Откройте консоль контроллера по SSH.
 - Остановите драйвер `wb-mqtt-serial` или иное ПО, которое опрашивает устройство.
3. Скачайте из репозитория файл прошивки для вашего устройства.
4. Загрузите файл прошивки на контроллер или другое устройство, на котором установлена утилита прошивки.
5. Перейдите в папку с файлом прошивки и прошейте устройство командой:

- на контроллере или компьютере с ОС Linux:

```
wb-mcu-fw-flasher -j -d /dev/ttyRS485-1 -a 25 -f ./firmware.wbfw
```

- на компьютере с ОС Windows:

```
wb-mcu-fw-flasher_1.0.3.exe -j -d COM1 -a 25 -f firmware.wbfw
```

6. Если вы выполняли команду с контроллера — запустите драйвер `wb-mqtt-serial`.

Здесь мы флагом `-j` переводим устройство, подключенное к порту `/dev/ttyRS485-1` (COM1) с адресом 25 в режим загрузчика и загружаем файл прошивки.

Успешный процесс прошивки выглядит так:

```
~# wb-mcu-fw-flasher -j -d /dev/ttyRS485-1 -a 25 -f mr6c_1.15.5_master_971fe50.wbfw
/dev/ttyRS485-1 opened successfully.
Send jump to bootloader command and wait 2 seconds...
Ok, device will jump to bootloader.
mr6c_1.15.5_master_971fe50.wbfw opened successfully, size 14720 bytes

Sending info block... OK

Sending data block 108 of 108... OK.
All done!
```

Если сигнатура устройства и файла прошивки не совпали, то вы получите сообщение об ошибке:

```
Sending info block...
Error while sending info block: Slave device or server failure
Data format is invalid or firmware signature doesn't match the device
```

Восстановление прошивки устройства

Если во время обновления произошел сбой, то устройство перейдет в режим загрузчика и вы можете восстановить его прошивку.

Автоматически

Для автоматического восстановления прошивки одного или нескольких устройств можно использовать утилиту `wb-mcu-fw-updater` в режимах **recover** и **recover-all**.

Чтобы восстановить устройство с адресом 10 и подключенное к порту `/dev/ttyRS485-1`, выполните команду:

```
wb-mcu-fw-updater recover /dev/ttyRS485-1 -a 10
```

Подробнее о режимах `recover` и `recover-all`, читайте в документации.

Вручную

Если вы не можете воспользоваться `wb-mcu-fw-updater`, то вы восстановить прошивку устройств можно с помощью сервисной утилиты `wb-mcu-fw-flasher`. Также этот способ могут использовать пользователи компьютеров с ОС Windows.

Для этого вам понадобится сама утилита и файл прошивки:

1. Подключите устройство по шине RS-485 к контроллеру или другому оборудованию, где установлена утилита прошивки.
2. Если вы выполняете команды на контроллере:
 - Откройте консоль контроллера по SSH.
 - Остановите драйвер `wb-mqtt-serial` или иное ПО, которое опрашивает устройство.
3. Скачайте из репозитория файл прошивки для вашего устройства.
4. Загрузите файл прошивки на контроллер или другое устройство, на котором установлена утилита прошивки.
5. Перейдите в папку с прошивкой и выполните команду:

- на контроллере или компьютере с ОС Linux:

```
wb-mcu-fw-flasher -d /dev/ttyRS485-1 -a 25 -f ./firmware.wbfw
```

- на компьютере с ОС Windows:

```
wb-mcu-fw-flasher_1.0.3.exe -d COM1 -a 25 -f firmware.wbfw
```

Здесь мы прошили находящееся в режиме загрузчика устройство с Modbus-адресом 25 и подключенное к порту `/dev/ttyRS485-1` (COM1) файлом `firmware.wbfw`.

Полезные ссылки

- Сброс Modbus-устройства Wiren Board к заводским настройкам
- Modbus-адрес устройства Wiren Board
- Утилита обновления и восстановления прошивок `wb-mcu-fw-updater`
- Сервисная утилита `wb-mcu-fw-flasher`
- Репозиторий прошивок для Modbus-устройств Wiren Board

Журнал изменений прошивок

Updating firmware

Please see this page for details. Firmware binaries are available on [fw-releases.wirenboard.com](http://fw-releases.wirenboard.com/?prefix=fw/) (<http://fw-releases.wirenboard.com/?prefix=fw/>).

Общая таблица по всем устройствам/ Summary table for all devices

Source project	Release date (YYYY-MM-DD)	Version	Affected devices	Changelog
WB-MR	2022-04-28	1.18.2 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.18.2/)	all	<ul style="list-style-type: none"> added: safety timer extension for targets - mr2m_k6, mrps6, mr6cu, mr6c_042, mr6cu_042, mr6_042, mrwl3_042, mr2mG, mr3G, mrps6G, mr6G, mr6cG, mrwl3G, mr6cuG, mrwm2G, mr6cpG, wbmwac, wbmwac_042, wbmwacG
WB-MR	2022-04-26	1.18.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.18.1/)	WB-MRWM2	<ul style="list-style-type: none"> added: input buttons support
WB-MR	2022-04-21	1.18.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.18.0/)	WB-MR	<ul style="list-style-type: none"> added: relay target with support for internal power supply mr6cpG (MR6Cv3), external power status register(holdreg 4) rework: added safety timer extension with setting the ability to control outputs(holdreg 930-936, 938-943) added: relay status setting: cause outputs to match inputs on power up
WB-MD	2022-04-25	2.5.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MD/main/2.5.0/)	WB-MDM3	<ul style="list-style-type: none"> added: PLL (phase-locked loop) feature: ability to support a wide range of input frequencies (such as 60 Hz) and noisy signals
WB-MR	2022-04-21	1.17.8 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.8/)	WB-MR	<ul style="list-style-type: none"> change: max debounce 250 ms
WB-MRGB	2022-04-03	3.0.4 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MRGB/main/3.0.4/)	WB-MRGBW-D	<ul style="list-style-type: none"> rework: Better press events handling, ability to disable double and shortlong events
WB-MR	2022-03-30	1.17.7 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.7/)	WB-MR	<ul style="list-style-type: none"> rework: Better press events handling, ability to disable double and shortlong events
WB-MR	2022-03-30	1.17.6 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.6/)	WB-MR	<ul style="list-style-type: none"> fixed: bootloader target names for GD32 targets
WB-MS	2022-03-24	4.18.6 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.18.6/)	WB-MSv2	<ul style="list-style-type: none"> added: support calibration of the light sensor for WB-MSv2 GD32 (holdreg 288)
WB-MRGB	2022-03-25	3.0.3 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MRGB/main/3.0.3/)	WB-MRGBW-D	<ul style="list-style-type: none"> added: PWM phase inversion between 1-2 and 3-4 channel (except 4*W mode)
WB-MD	2022-03-18	2.4.2 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MD/main/2.4.2/)	WB-MDM3	<ul style="list-style-type: none"> added: eeprom "credits" for status saving, credits increases every 300s, status saves every 1s if there are credits
WB-MRGB	2022-03-15	3.0.2 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MRGB/main/3.0.2/)	WB-MRGBW-D	<ul style="list-style-type: none"> fixed: random bursts on VDD on GD32 targets
WB-MAO4	2022-03-15	2.1.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MAO4/main/2.1.1/)	WB-MAO4	<ul style="list-style-type: none"> fixed: random bursts on VDD on GD32 targets
WB-MR	2022-03-05	1.17.5 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.5/)	WB-MR, WB-MRWM2	<ul style="list-style-type: none"> fixed: jumps in power readings during load disconnection due to frequency measurement errors on the MRWM2 and frequency measurement errors at the inputs ERRMR09
WB-MRGB	2022-03-10	3.0.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MRGB/main/3.0.1/)	WB-MRGBW-D	<ul style="list-style-type: none"> fixed: Make GD32 target working
WB-MAI	2022-03-01	1.3.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MAI/main/1.3.1/)	WB-MAI11	<ul style="list-style-type: none"> fixed: algorithm of saving common settings in EEPROM
WB-MS	2022-02-	4.18.4 (http://fw-rel	WB-MS	

	25	eases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.18.4/)		<ul style="list-style-type: none"> fixed: Check write completed on power down. Add delay for capacitor discharging on power down
WB-MR	2022-02-18	1.17.4 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.17.4/)	WB-MRWM2	<ul style="list-style-type: none"> added: support for diagnostic registers(input 368-371) showing the frequency of the signal at the output of the zero detector
WB-MAI	2022-02-18	1.3.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAI/main/1.3.0/)	WB-MAI11	<ul style="list-style-type: none"> add: Measuring sampling period of each channel fixed: More accuracy lowpass filter: time constant is calculated for each channel based on its sampling period ERRMAI110005
WB-MAI	2022-02-15	1.2.6 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAI/main/1.2.6/)	WB-MAI11	<ul style="list-style-type: none"> add: Put error value to modbus immediately after channel mode was changed. Then error value will be replaced with true data after first measurement fixed: Restart channel measurement if it's settings was changed while measurement fixed: First measure special channels (such as AVCC, ATEMP), then data channels. This produced incorrect first measurement if AVCC or ATEMP used in calculations fixed: Reset lowpass filter when gain is changed automatically ERRMAI110004 fixed: Use repetition count in self-heating compensation formula
WB-MD	2022-02-15	2.4.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.4.1/)	WB-MDM3	<ul style="list-style-type: none"> fixed: Overcurrent protection handling is available only for "mdm3_26" and "mdm3G26" signatures
WB-MCM	2022-02-02	1.3.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MCM/main/1.3.2/)	WB-MCM8	<ul style="list-style-type: none"> fix: the error of no response on modbus added: input buttons counter support added: support save to flash storage for buttons time settings added: input mode support (holdreg 9-16)
WB-MD	2022-02-14	2.4.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.4.0/)	WB-MDM3	<ul style="list-style-type: none"> added: Overcurrent protection handling: disable outputs for 3 s if OCP triggered ERRMDM01 added: Holdreg 100: OCP status (0 - normal; 1 - triggered)
WB-MD	2022-02-11	2.3.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.3.3/)	WB-MDM3	<ul style="list-style-type: none"> fixed: FETs fully opened if raw_duty is less than 220 on trailing edge or 320 on leading edge ERRMDM06 added: Minimum rise/fade time is limited on 1ms/% added: Soft-start feature: raw_duty smoothly increases form 0 to min_duty when enabling ERRMDM01
WB-MS	2022-02-10	4.18.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.18.3/)	WB-MS	<ul style="list-style-type: none"> fix: adc stop when erase ERRWB-MS0011
WB-MAI	2022-02-09	1.2.5 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAI/main/1.2.5/)	WB-MAI11	<ul style="list-style-type: none"> fixed: incorrect State value of the input module in the "dry contact" mode ERRMAI110003
WB-MR	2022-02-09	1.17.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.17.3/)	WB-MRM2-mini old, WB-MRWM2	<ul style="list-style-type: none"> fixed: make firmware
WB-MD	2022-02-07	2.3.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.3.2/)	WB-MDM3	<ul style="list-style-type: none"> fixed: Dimming curve interpolation on range edges ERRMDM05
WB-MS	2022-01-26	4.18.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.18.2/)	WB-MS	<ul style="list-style-type: none"> fix: CO2 sensor range configure ERRWB-MSWv30010
WB-MAP	2022-01-24	2.3.7 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.7/)	all	<ul style="list-style-type: none"> added: MCU internal voltage and temperature registers added: minimum input voltage register
WB-MAP	2022-01-12	2.3.6 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.6/)	MAP3E MAP6S	support new FRAM chips
WB-MR	2022-01-26	1.17.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.17.2/)	all	<ul style="list-style-type: none"> fix ERRMR08: relay power pwm update latency

WB-MR	2022-01-31	1.17.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.1/)	WB-MRM2-mini	<ul style="list-style-type: none"> added: input buttons support for WB-MR2-mini
WB-MR	2021-12-21	1.17.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.17.0/)	WB-MR	<ul style="list-style-type: none"> added: input buttons support added: additional mapping matrix with the ability to configure inputs as buttons for detecting various types of clicks
WB-MD	2022-02-02	2.3.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MD/main/2.3.1/)	WB-MDM3	<ul style="list-style-type: none"> fixed: Short-term load switching when VIN enabling in 50% cases fixed: Half-periods are sometimes skipping when raw duty is around 1000 us and trailing edge fixed: CH2 and CH3 are not working in switch mode if value of CH1 (holdreg 0) is 0 fixed: If CH1 in switch mode and value of it's holdreg is changed to 0 from enabled state, the load actually not disable fixed: Make modbus more stable on high bauds
WB-MS	2022-01-31	4.18.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.18.1/)	WB-MS	<ul style="list-style-type: none"> added: independent of stop bit settings, holdreg 112 is ignored
WB-MCM	2022-01-28	1.3.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MCM/main/1.3.1/)	WB-MCM8	<ul style="list-style-type: none"> fix power down counters save ERRMCM01 fix V_MCU T_MCU ERRMCM02
WB-MRGBW-D	2022-01-28	3.0.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MRGBW/main/3.0.0/)	WB-MRGBW-D	<ul style="list-style-type: none"> added: New button controls mechanism. Short, long, double, shortlong event handlers added: Use flash_storage for settings saving rework: Use wb_rcc added: 11 dimmer modes added: CTT support added: RGB <-> HSV conversion added: Hue changing function added: Counters for each click types (short, long, etc) added: 320-323 holdregs stores version as digits: major, minor, patch, suffix added: 324-325 holdregs stores version as uint32 in little-endian format added: 326-327 holdregs stores version as uint32 in big-endian format
WB-MS	2022-01-28	4.18.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.18.0/)	WB-MSWv3, WB-MIR	<ul style="list-style-type: none"> added: Holdreg 5500 - play IR command from ROM added: Holdreg 5501 - edit IR command (ROM -> RAM) added: Holdreg 5502 - learn IR command to ROM fixed: Reset all ROMs command (coil 5000) reset only first ROM ERRMIR04 (https://wirenboard.com/wiki/WB-MIR_v2:_Errata#ERRMIR04:_По_команде_Reset_All_ROMs_стирается_только_ROM1) fixed: ROM Size is not updated if ROM was cleared by editing command fixed: Error is returned when coil disabled after editing ROM if first two regs are zeroes
WB-MS	2022-01-27	4.17.7 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.17.7/)	WB-MSWv3	make target MSW3_4_9_GD32_TH
WB-MS	2022-01-26	4.17.6	all	improve adc driver. fix random bursts in adc channels like PIR or SPL.
WB-MS	2022-01-26	4.17.5	WB-MAI2-mini	added registers (holdreg 273,274) for setting the low-pass filter for inputs and saving setting to EEPROM

Прошивки, выпущенные после 2022-01-26 доступны для обновления только с помощью wb-mcu-fw-updater версии 1.1.1 и выше (входит в релиз wb-2201)

Firmwares released after 2022-01-26 available for upgrade only with wb-mcu-fw-updater version 1.1.1 or above (included in wb-2201 release)

WB-MS	2022-01-19	4.17.4 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.17.4/)	WB-MSWv3	<p>increase in the measurement speed due to the fact that the illumination value is written</p> <p>to the register at each measurement of the light sensor WB-MSW v.3 hw. 4.19.</p>
WB-MS	2022-01-14	4.17.3 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.17.3/)	WB-MSWv3	added support calibration of the light sensor WB-MSW v3 hw. 4.19 (holdreg 288)
WB-MR	2021-12-13	1.16.4 (http://fw-releases.wirenboard.com/?prefix=fw/by-v)	WB-MR	added support MRWM2 voltage and power measure relay module

		ersion/WB-MR/main/1.16.4/)		
WB-MS	2021-12-14	4.17.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.17.2/)	WB-MSWv3	increase in measurement speed for light sensor WB-MSW v3 hw. 4.19
WB-MS	2021-12-03	4.17.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.17.1/)	WB-MSWv3	new lid transmittance constant for WB-MSW v.3 hw. 4.19
WB-MS	2021-11-22	4.17.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.17.0/)	WB-MSWv3	support for WB-MSW v.3 hw. 4.19
WB-MAP	2021-11-30	2.3.5 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.5/)	all	fix: power fail level = 3.8 v
WB-MAP	2021-11-30	2.3.4 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.4/)	MAP12E	add target MAP12E GD32
WB-MAP	2021-11-30	2.3.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.3/)	WB-MAP3E	add target MAP3E GD32
WB-REF-CR	2021-11-13	1.0.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-REF-CR/main/1.0.1/)	WB-REF-U-CR	Fix modbus device signature
WB-MR	2021-10-27	1.16.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.16.3/)	WB-MR	<ul style="list-style-type: none"> fix inputs frequency measurement ERRMR06 fix the error of no response on modbus ERRMR07
WB-MRGBW-D	2021-10-25	1.3.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MRGBW/main/1.3.2/)	WB-MRGBD-W	<ul style="list-style-type: none"> fixed status saving when power fall by decreasing clock speed added "credits" for status saving: credits increases every 300s, status saves every 1s if there are credits
WB-MS	2021-10-04	not released	WB-MS, WB-MSWv3	<ul style="list-style-type: none"> rework timemanager and i2c driver fix voc sensors ERRWB-MS0008 fix spl autocalibration ERRWB-MSWv30006 fix pir freeze ERRWB-MSWv30007
WB-REF-CR	2021-09-13	1.0.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-REF-CR/main/1.0.0/)	WB-REF-U-CR	First public release
WB-REF-DF	3-09-2021	1.0.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-REF-DF/main/1.0.1/)	WB-REF-DF-178A	<ul style="list-style-type: none"> fix software reset ERRWB-REF-DF0001. fix no modbus error response when reading with function 0x04(Read Input Registers) ERRWB-REF-DF0002.
WB-MS	30-08-2021	4.16.17 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.17/)	all	fix software reset ERRWB-MS0008
WB-REF	27-08-2021	1.0.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-REF-DF/main/1.0.0/)	WB-REF-DF-178A	add support for danfoss refrigeration controller for EKC 202B, EKC 202D, EKC 204A1
WB-MS	23-08-2021	4.16.16 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.16/)	MSv2, MSWv3	<ul style="list-style-type: none"> add user temp compensation register MSv2 (holdreg 245) and registers of raw values of the temperature and humidity sensor for MSv2 and MSWv3 (holdreg 284 and 285) add dynamic calculation of temperature compensation for devices with CO2 and VOC sensor
WB-MAP	2021-04-29	2.3.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.2/)	WB-MAP3E	support for WB-MAP3E hw. rev.1.3
WB-MAP	2021-03-17	2.3.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.1/)	WB-MAP3E, WB-MAP6S	<p>support for customer-specific WB-MAP3E model (WB-MAP3E-36A)</p> <p>fix reporting of negative power on WB-MAP6S</p>

		rsion/WB-MAP/main/2.3.1/)		
WB-MAP	2020-12-08	2.3.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MAP/main/2.3.0/)	WB-MAP*	major refactoring. Support for phases remapping on WB-MAP3E and WB-MAP12E
WB-MAP	2020-12-07	2.2.8 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MAP/main/2.2.8/)	WB-MAP*	The same as 2.2.6, reverting 2.2.7
WB-MS	12-08-2021	4.16.15 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.15/)	WB-MS	<ul style="list-style-type: none"> add target M1W2 GD32 fix MCU internal voltage and temperature registers on GD32 based devices ERRWB-MS0006 fix impulse counter M1W2 and VOC baseline MSWv3 save in power fail ERRWB-MS0007
WB-MAI	2021-07-28	1.2.4 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MAI/main/1.2.4/)	WB-MAI11	improve accuracy of 2W resistance measurements by 0.08 Ohm
WB-MR	28-07-2021	1.16.2 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.16.2/)	WB-MWAC	fix counters zero values in holdregs during 1 sec after boot ERRMWAC01
WB-MR	28-07-2021	1.16.1 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.16.1/)	WB-MR	<ul style="list-style-type: none"> fix MCU internal voltage and temperature registers on GD32 based devices ERRMR04 fix coils status save in power fail ERRMR05
WB-MD	26-07-2021	2.3.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MD/main/2.3.0/)	WB-MDM3	<ul style="list-style-type: none"> GD32 target Add MCU internal voltage and temperature registers Add minimum input voltage register FIX: status save intervals. ERRMDM03
WB-MS	08-07-2021	4.16.14 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.14/)	WB-MSW v.3	target for MSW v3 TH without SPL and PIR
WB-MS	07-07-2021	4.16.13 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.13/)	WB-MS v.2	fix illumination measurement ERRWB-MSv20001.
WB-MS	5-07-2021	4.16.12 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.12/)	WB-MSW v.3	fix start motion sensor MSWv3 ERRWB-MSWv30005.
WB-MS	29-06-2021	4.16.11 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.11/)	WB-MSW v.3	fix synchronization of illumination measurement and LED switching on ERRWB-MSWv30003.
WB-MR	21-06-2021	1.16.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.16.0/)	WB-MR	<ul style="list-style-type: none"> GD32 target Add MCU internal voltage and temperature registers Add minimum input voltage register
WB-MCM	15-06-2021	1.3.0 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MCM/main/1.3.0/)	WB-MCM8	<ul style="list-style-type: none"> GD32 target Add MCU internal voltage and temperature registers Add minimum input voltage register
WB-MR	28-05-2021	1.15.6 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.15.6/)	WB-MWAC	fix WB-MWAC specific functions for STM32F042K6 target
WB-MR	25-05-2021	1.15.5 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MR/main/1.15.5/)	WB-MRWL3	target for MRWL3 on STM32F042K6 chip
WB-MS	24-05-2021	4.16.9 (http://fw-releases.wirenboard.com/?prefix=fw/by-version/WB-MS/main/4.16.9/)	WB-MSW v.3, WB-MIR v2	<ul style="list-style-type: none"> WB-MSW v.3 mic curves fix fix IR bank change, when all IR banks used. ERRMIR02 GD32 fix adc when flash erase
WB-MS	18-05-2021	4.16.8 (http://fw-releases.wirenboard.com/?prefix=fw/by-v	all	rework startup. GD32 support.

		ersion/WB-MS/main/4.16.8/)		
WB-MR	14-05-2021	1.15.4 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.15.4/)	WB-MR6	target for MR6 on STM32F042K6 chip
WB-MAI	08-05-2021	1.2.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAI/main/1.2.3/)	WB-MAI11	fix 50 day freeze ERRMAI110002.
WB-MS	05-05-2021	4.16.7 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.7/)	WB-MSW v.3, WB-MS v.2, WB-MIR v2, WB-M1W2, WB-MAI2mini	fix 50 day freeze
WB-MR	05-05-2021	1.15.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.15.3/)	all	<ul style="list-style-type: none"> Target for STM32F042K6 fix 50 day freeze ERRMR03
WB-MD	05-05-2021	2.2.4 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.2.4/)	WB-MDM3	fix 50 day freeze ERRMDM02
WB-MD	28-04-2021	2.2.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.2.3/)	WB-MDM3	Target for STM32F042K6
WB-MS	15-04-2021	4.16.6	WB-MSW v.3	<ul style="list-style-type: none"> New MSW3's lid transmittance constant.
WB-MAI	05-04-2021	1.2.2	WB-MAI11	Improve accuracy for 3-wire resistance measurement. Fixes ERRMAI110001.
WB-MS	08-02-2021	4.16.5 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.5/)	WB-MSW v.3, WB-MS v.2, WB-MIR, WB-M1W2, WB-MAI2mini	<ul style="list-style-type: none"> Fixed uart freezing on a noisy line with ongoing communication at 115200 baud rate.
WB-MS	04-02-2021	4.16.4 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.4/)	WB-MSW v.3	<ul style="list-style-type: none"> Add SPL calibration data for MEMS mic.
WB-MS	01-02-2021	4.16.3 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.3/)	WB-MSW v.3	<ul style="list-style-type: none"> New MSW model target with only hdc1080 sensor and buzzer.
WB-MR	24-12-2020	1.15.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.15.2/)	all	fix safety timer (problem in 1.15.0, 1.15.1)
WB-MS	21-12-2020	4.16.2 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.2/)	WB-MSW v.3, WB-MS v.2, WB-MIR, WB-M1W2, WB-MAI2mini	<ul style="list-style-type: none"> Fixed unstable modbus communication under wb-mqtt-serial fast polling condition.
WB-MD	04-12-2020	2.2.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.2.1/)	WB-MDM3	<ul style="list-style-type: none"> Fix zero cross time, considers FET close time.
WB-MCM	02-12-2020	1.2.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MCM/main/1.2.0/)	WB-MCM8	<ul style="list-style-type: none"> Add inputs debounce parameters. The default value is 50 ms, which can be changed by Modbus master, saved in EEPROM. Add inputs frequency calculation.
WB-MS	20-11-2020	4.16.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.1/)	WB-MSW v.3	<ul style="list-style-type: none"> Add dynamic temperature compensation for devices with onboard CO2 sensor. Substructured factory hardcoded temperature compensation parameter. Now temperature compensation is applied only when CO2 sensor is operating and 245 register is left for user temperature adjustments Fix illuminance sensor work at high illumination conditions
WB-MS	23-10-2020	4.16.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.16.0/)	WB-MSW v.3, WB-MS v.2, WB-MIR, WB-M1W2, WB-MAI2mini	<ul style="list-style-type: none"> Add MCU internal voltage and temperature registers Add minimum input voltage register Fix unstable modbus communication on 115200 baudrate Fix unstable co2 sensor communication Fix m1w2 unstable digital input mode
WB-MAP	10-10-2020	2.2.7 (http://fw-releases.wireboard.com/)	MAP6SE, MAP3E, MAP12E	<ul style="list-style-type: none"> Target for MAP6SE.

		m/?prefix=fw/by-version/WB-MAP/main/2.2.7/)		<ul style="list-style-type: none"> Delete not existed regs in E devices.
WB-MS	07-10-2020	4.15.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.15.1/)	WB-MSW v.3	<ul style="list-style-type: none"> affect: WB-MSW v.3 rev 4.9 note: Improved SPL accuracy for low dB range for some sensors. note: Report measurements outside well-defined response curves. The total range of reported values is 37.4-115 dBA
WB-MS	28-09-2020	4.15.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.15.0/)	ALL	<ul style="list-style-type: none"> Improve input voltage measure and powerdown handle Increase software I2C speed to 50 kHz for faster EEPROM writing Fixed modbus frame borders detection Added holding register 113 to set modbus inter frame timeout Rework hdc1080, opt3001, sgpc3 modules with new non blocking i2c library Added MSW v3 rev 4.9 target with mems mic and additional highgain adc input channel Removed SPL linear approximation calculation. All targets use response tables Changed digital input counter saving to EEPROM algorithm for devices with digital inputs. Previously counters data was loaded to EEPROM once per 600 sec. Now if counter increments slower than 1 time per 300s, the data uploads in EEPROM for each change, otherwise, not faster than 1 time per 300s.
WB-MS	14-09-2020	4.14.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.14.1/)	ALL	Reduce 1wire sensors initialization time.
WB-MR	02-09-2020	1.15.1 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.15.1/)	WB-MIR WB-M1W2 WB-MSv2	Improve input voltage measure and powerdown handle.
WB-MS	31-08-2020	4.14.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.14.0/)	ALL	Rework all sensors with task manager module
WB-MR	06-08-2020	1.15.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MR/main/1.15.0/)	ALL	Add inputs frequency measurement
WB-MS	26-06-2020	4.13.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.13.0/)	WB-MIR WB-MIR64 WB-M1W2 WB-M1W2_V2_1	Fix compensation internal ntc temperature sensor.
WB-MS	23-04-2020	4.12.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.12.0/)	WB-MSW v.3	Improve TH sensor work. Read errors filtration.
WB-MD	04-04-2020	2.2.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MD/main/2.2.0/)	WB-MDM3	Add switch mode, mode selection by hold reg 50-52 (value 2)
WB-MS	01-04-2020	4.11.0 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MS/main/4.11.0/)	WB-MSW v.3	Improve CO2 sensor work. Read settings from sensor.
WB-MD	19-03-2020	2.1.0	WB-MDM3	Two modbus holding registers 140 and 150 were added for setting the variable dimming duration
WB-MAP	10-03-2020	2.2.5 (http://fw-releases.wireboard.com/?prefix=fw/by-version/WB-MAP/main/2.2.5/)	MAP12H,MAP3E,MAP3H,MAP6S	Fix FRAM configuration loss
WB-MS	19-03-2020	4.10.0	WB-M1W2	M1W2 v1.2 with active pullup support
WB-MIO	24-12-2019	1.5.1	ALL	Change build system use make
WB-MR	25-11-2019	1.14.1	ALL	Change build system use make
WB-MR	25-11-2019	1.14.0	WB-MR6CU	New model MR6CU compact 2 unit 6 channel 7A relay without inputs
WB-MAP	14-11-2019	2.2.2	MAP6S	Target for STM32F051K6
WB-MS	12-11-2019	4.9.0	WB-MSW v.3	<ul style="list-style-type: none"> New calibration data for spl-meter Automatic sound baseline calibration to account for opamp offset Add spl offset register

WB-MR	01-11-2019	1.13.1	WB-MR2mini	Fix input mode default value. add led in WB-MR2mini v2.1
WB-MR	18-10-2019	1.13.0	WB-MR2mini, WB-MR3, WB-MR6, WB-MR6C, WB-MWAC	Variable debounce 0-100ms, reg 20+
WB-MRGB	25-09-2019	1.3.0	WB-MRGBW-D	Work with bootloader 1.1.0 - support hw watchdog.
WB-MAP	10-09-2019	2.2.0	WB-MAP3E, WB-MAP3H, WB-MAP6S, WB-MAP12H	Eeprom and perith submodules. RAM optimisation. Work with bootloader.
WB-MCM	27-09-2019	1.1.0	WB-MCM8	Work with bootloader 1.1.0 - support hw watchdog.
WB-MS	19-09-2019	4.8.0	ALL	Work with bootloader 1.1.0 - support hw watchdog. use submodules.
WB-MS	06-09-2019	4.7.0	WB-MSWv3	Added STM32F030 mcu. IR module disabled. Use MS bootloader target.
WB-MIO	15-08-2019	1.5.0	ALL	Bootloader support
WB-MS	13-06-2019	4.6.1	WB-V2	Added support of MS v2 sensor with another coefficients for adc to measure lux using OSRAM_BPW34S sensor.
WB-MR	13-06-2019	1.12.0	WB-MR*, WB-MWAC	Default input mode switch (1)
WB-MCM	28-05-2019	1.0.0	WB-MCM8	Initial firmware version: 32-bit EEPROM-stored counters; digital inputs LED indication
WB-MR	17-05-2019	1.11.1	WB-MR*, WB-MWAC	Fix invalid inputs state in discrete registers after startup
WB-MS	22-03-2019	4.6.0	WB-MIR, WB-M1W2	Added w1 temperature registers without invalid state - it save previous valid (20 - 21) added w1 channels status discret regs (16 - 17)
WB-MS	04-03-2019	4.5.0	WB-MS, WB-MSW v.3, WB-MIR, WB-M1W2, WB-MAI2-mini/cc	Support firmware update
WB-MS	27-02-2019	4.4.0	WB-MIR	Mir64 version with 40 ir codes cells
WB-MS	11-02-2018	4.3.0	WB-MIR	More robust IR commands storage (i.e. without flash fs and compression), as in fw < 3.7.2
WB-MS	11-02-2019	4.2.0	WB-MSW v.3	* Added: support for MSW v.3 hw rev 4.8 * added: temperature and relative humidity x100 value to 4 and 5 registers * added: temperature and relative humidity self-heat compensation 245 register 1x100 *C * added: new register 108: SGPC3 sensor version. 0xFFFF is sensor is missing on power-up * change: sgpc3: ignoring data during 3 minutes after warm up (total 364 seconds after power up).
WB-MR	04-03-2019	1.10.0	WB-MR*, WB-MWAC	Support firmware update -
WB-MRGB	2019-03-04	1.2.0	WB-MRGB-D	Support firmware update
WB-MR	2019-02-14	1.9.4	WB-MR*, WB-MWAC	* Change: fix change modbus id via broadcast 0 address
WB-MR	2018-11-14	1.9.2	WB-MR*, WB-MWAC	Add check valid for readed from eeprom settings Add check valid for modbus address when changed via modbus and when readed from eeprom
WB-MR	2018-11-14	1.9.1	WB-MR*, WB-MWAC	<ul style="list-style-type: none"> ▪ More robust configuration storage in EEPROM ▪ Change: I2C EEPROM ic is used to store basic configuration. ▪ Added: new input-output relationship handling is implemented: there is a new input mode which tells the fw to use so called input mappings to decide what to do on input state change. This mapping, distinct for each input-output pair, allows to set actions for both rising and falling edges of input signal. The actions are: set output, reset output, toggle output, do nothing. ▪ Change: Kill-switch function is basically removed. It replaced with simplified input mode 2 which switches off all output channels on rising edge of the signal. ▪ Added: WB-MWAC water leak controller is supported
WB-MRGB	2019-02-13	1.1.3	WB-MRGBW-D	* Change: fix change modbus id via broadcast 0 address (fixes ERRMRGBWD0001)

WB-MAP	2019-02-03	2.1	WB-MAP3H, WB-MAP3E, WB-MAP12H, WB-MAP6S	<p>* Change: WB-MAPs and CTs are now supposed to be calibrated separately.</p> <p>Each CT is described by two parameters: (effective) turns ratio and phase delay.</p> <p>These parameters are to be programmed into WB-MAP registers to proper operation</p> <p>* Change: phase angle is consistent between MAP3E and other models (-180..+180 notation)</p> <p>* Added: 32-bit registers for voltage and current</p>
WB-MRGB	2018-12-05	1.1.2	WB-MRGBW-D	<p>* Fix change modbus uart settings</p> <p>* Change eeprom files to submodule. add necessary defines. change project paths</p> <p>* Move eeprom settings load/save to separate file.</p> <p>* Move eeprom settings struct defines from config.h to settings.c.</p> <p>* Add modbus id change validation</p> <p>* Add validation modbus settings when load from eeprom.</p> <p>* Add validation for buttons disable, pwm divider and fade time settings</p> <p>* Disable 1200 baud variant (need research why not work)</p>
WB-MDM2		1.1.0	WB-MD2	Two modbus holding registers 65 and 66 were added for users can choose 1 of 3 dimming curves: (0)incandecent bulbs, (1)LED bulbs, (2)resistive load
WB-MS		4.1.0	WB-MSW v.3	<p>* Note: MSW v.3 VOC-sensor related fixes and improvenets</p> <p>* Added: input register 106 with current valid SGPC3 baseline reported by the sensor</p> <p>* Added: input register 107 with current raw signal</p> <p>* Change: VOC sensor is initialized for 184s after power-on. During this time VOC registers return error value.</p>
WB-MS		4.0.1	WB-MSW v.3	* Added: improve SPL metering on WB-MSW v.3
WB-MS		4.0.0	WB-MS, WB-MSW v.3, WB-MIR, WB-M1W2, WB-MAI2-mini/cc	<p>* Added: add new target for WB-MSW v3</p> <p>* Added: add PIR movement sensor support</p> <p>* Added: add SGPC3 air quality sensor support</p> <p>* Added: add discrete input mode for 1-wire inputs with activation counters</p> <p>* Added: improve SPL metering on WB-MS</p> <p>* Change: improved config storage</p> <p>* Fixes: workaround for IR bug</p>
WB-MS		3.12.2	WB-MAI2-mini/CC	Add support for WB-MAI2-mini/CC
WB-MS		3.12.1	WB-MIR, WB-M1W2	<p>* Change: fixes NTC heating compensation</p> <p>* Note NTC compensation value was damaged while saving/restoring from flash</p>
WB-MS		3.12	WB-MSW2	<p>* Change: fixes NTC heating compensation</p> <p>* Change: add <censored> new CO2 sensor support to MSW2_3.4 boards</p> <p>* Fixes modbus integrity check</p> <p>* Checklist:add manual calibration for <censored></p> <p>* Checklist:add zero calibration (manual calibration to 400ppm)</p> <p>* Note: - Write 1 to coilreg (COIL_REG_CO2_SENS_CALIBRATE_ZERO) 1 to fresh air calibrate any CO2 sensor (At <censored> the 1 value at coilreg remains 1 for 3 sec and then = 0)</p> <p>- Write 1 to holdreg (HOLD_REG_CO2_SENS_ABC_CALIBRATION) 95 to CLOSE CO2 sensor ABC calib/ 0 = OPEN At changing the parameter ABC cycle is also transmitted to sensor</p> <p>- Write any value between 400-1500 into () 88 to manually calibrate <censored> sensor Register content is automatically set to 0 after calibration.</p> <p>- Write 1-15 to holdreg () 89 to set ABC cycle (days). At setting the register OPEN/CLOSE state is also transmitted.</p>

WB-MS		3.11.2	WB-MS, WB-MSW, WB-MIR, WB-M1W2	* Change: bug fix: writing single hold reg value > 125 resulted modbus illegal data value error
WB-MS		3.11.1	WB-MS, WB-MSW, WB-MIR, WB-M1W2	<p>* Change: add modbus package integrity testing to "mb_recive_hadler" function</p> <p>* Note: - returns with modbus error "MB_ERROR_ILLEGAL_DATA_VALUE" if a package includes a write CRC but wrong package size or fields - returns with modbus error "MB_ERROR_ILLEGAL_DATA_VALUE" if requested size of data is greater than allowed (125 at reading, 123 at writing but at writing technically not possible to get receive such command due to the limited RX buffer size) - returns with modbus error "MB_ERROR_ILLEGAL_FUNCTION" if modbus request function is unknow.</p>
WB-MS		3.10.1	WB-MS, WB-M1W2, WB-MIR	1-wire reset time changed from ~410 us to ~550 us. As in 1-Wire Standard, should be between 480 and 640 us
WB-MS		3.10.0	WB-MS, WB-M1W2, WB-MIR	Added filter out algorithm for suspicious values 85C and 127.937C from 1-wire temperature sensors
WB-MRGB		1.1.1	WB-MRGBW-D	<ul style="list-style-type: none"> ▪ Storing configs and device state in external eeprom ▪ Watchdog enable ▪ Change: add MRGBW support <p>* Note: modbus hold reg 3 = white channel value modbus hold reg 8 = button 3 value button3 short press = on/off white channel button3 long press = adjust brightness of white channel modbus hold reg 33 = button 3 counter</p> <ul style="list-style-type: none"> ▪ Change: "BUTTON_DISABLED" register state is stored/restored to eeprom ▪ Change: Effectless "color changed over modbus" feature removed
WB-MAP		1.1	WB-MAP3H, WB-MAP3E, WB-MAP6S, WB-MAP12H	Add support for WB-MAP3 devices

Retrieved from "<https://wirenboard.com/wiki/Служебная:Print/>"

- Privacy policy
- About Wiren Board
- Disclaimers
-