



**МОДУЛЬ
АВТОМАТИКА**

Коммуникационный модуль PL307

Руководство по эксплуатации
ПЛАБ.421000.027 РЭ



г. Пенза

2019

Содержание

Содержание.....	2
Введение	3
1 Описание и работа изделия.....	4
1.1 Назначение изделия.....	4
1.2 Состав исполнений.....	5
1.3 Структурная схема.....	5
1.4 Технические характеристики	5
1.5 Устройство и работа.....	7
2 Настройка режимов работы	12
2.1 Установка связи с модулем.....	12
2.2 Настройка режима работы.....	13
2.3 Настройка параметров связи порта Ethernet.....	16
2.4 Настройка RS-485.....	20
2.5 GSM-сеть.....	20
2.6 Таблица опроса	21
2.7 Мониторинг данных	25
2.8 Журнал событий	30
2.9 Пароли.....	33
2.10 Выполнение скриптов	34
2.11 Логи модуля.....	37
2.12 Обновление прошивки модуля.....	38
3 Создание динамических сайтов.....	39
3.1 Общие сведения	39
3.2 Создание сайта.....	39
3.3 Текст файла « <code>ajax.js</code> »	41
3.4 Пример написания javascript для динамического обновления переменных на странице сайта	42
3.5 Работа с архивными данными	43
3.6 Загрузка сайта в модуль PL307	45
4 Техническое обслуживание.....	46
4.1 Общие указания.....	46
4.2 Меры безопасности	46
4.3 Порядок технического обслуживания изделия	46
4.4 Замена батарейки часов	46
5 Хранение и транспортировка	47
5.1 Транспортирование	47
5.2 Хранение	47
6 Гарантийные обязательства изготовителя	48

Введение

Настоящее руководство по эксплуатации содержит описание, устройство, технические характеристики, базовые принципы практического использования, правила хранения и текущего обслуживания, а также другие сведения, позволяющие реализовать в полном объеме технические возможности Коммуникационного модуля PL307 (далее «Модуль» или «Модуль PL307»). Перед началом эксплуатации устройства необходимо внимательно ознакомиться с настоящим документом.

К работе с изделием допускается квалифицированный персонал, имеющий необходимые навыки работы с изделием.

1 Описание и работа изделия

1.1 Назначение изделия

Модуль PL307 содержит интерфейсы Ethernet, RS-485, опционально GSM, а также слот для установки карты памяти microSD. Модуль предназначен для выполнения следующих функций:

встроенный веб-сервер – позволяет построить визуальный человеко-машинный интерфейс для контроля и управления оборудованием; для работы достаточно обычного браузера, запущенного на персональном компьютере, смартфоне и аналогичных устройствах;

сервер сбора данных (концентратор данных). Этот режим работы позволяет вести непрерывный опрос внешних устройств по протоколам Modbus RTU и Modbus TCP, поддерживая в памяти модуля актуальную базу текущих значений переменных, получаемых с внешних устройств, и быстро отдавать их по запросу;

архивирование любых данных, получаемых с Modbus-устройств с сохранением в виде «срезов» на карту памяти microSD и возможностью последующего чтения архивных данных по протоколу HTTP (включая визуализацию в виде графиков посредством встроенного веб-сервера) и FTP (в виде файлов);

регистрация событий, связанных с изменением значений переменных, с ведением журнала и возможностью отправки уведомлений по E-mail и SMS¹ (для исполнений со встроенным GSM-модемом);

шлюз Modbus TCP – Modbus RTU – может работать в двух режимах и выполняет преобразование Modbus TCP в Modbus RTU, либо Modbus RTU в Modbus TCP;

подключение в качестве клиента к внешнему серверу мониторинга, с которого выполняется контроль и управление объектом с помощью специализированного ПО.

Контроллер предназначен для эксплуатации при следующих условиях окружающей среды:

рабочий температурный диапазон, °Сот 0 до 60 (опц. от -20 до 60);
относительная влажность воздуха (при 25 °С), %, не более..... 85;
атмосферное давление, кПа (мм рт.ст.)от 84,0 до 106,7 (от 630 до 800).



Рисунок 1.1 – Внешний вид модуля

¹ Разработка

1.2 Состав исполнений

Наименование	Ethernet	RS-485	GSM
PL307-1RS	1x 10/100 Base-T/TX	1 канал	-
PL307-3RS ¹	1x 10/100 Base-T/TX	3 канала	-
PL307-1GM1RS	1x 10/100 Base-T/TX	1 канал	2G

1.3 Структурная схема

Структурная схема показана на рисунке ниже.

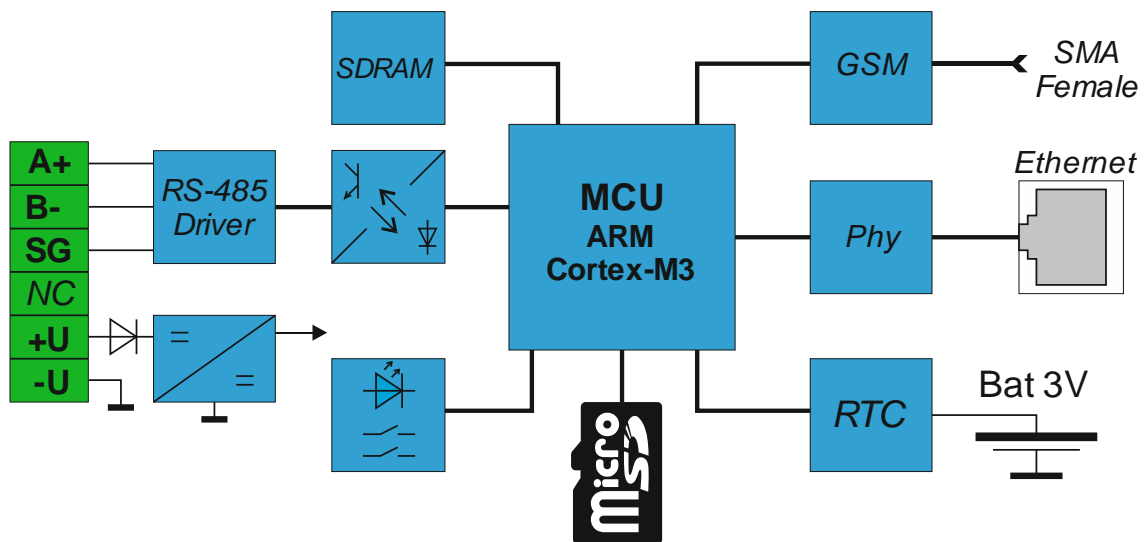


Рисунок 1.2 – Структурная схема

1.4 Технические характеристики

1.4.1 Основные технические характеристики

Номинальное напряжение питания, В (DC)	24;
Допустимый диапазон напряжения питания, В	от 18 до 30;
Потребляемая мощность, Вт, не более	1,2;
Наработка на отказ, ч	75 000;
Срок службы устройства, лет, не менее	7;
Габаритные размеры, мм (ШxВxГ)	36,6x96x58;
Вес, г, не более	100 г;
Степень защиты оболочки	IP30;
Монтаж	на DIN-рейку.

1.4.2 Параметры портов RS-485

Диапазон скоростей передачи данных, бод	от 1 200 до 256 000;
---	----------------------

Встроенные терминирующие резисторы Нет;
 Напряжение гальванической изоляции, В (АС) 1500;
 Защита сигнальных линий супрессоры 6V;

1.4.3 Параметры порта Ethernet

Режимы работы 10BASE-T/100BASE-TX;
 Поддержка HP Auto-MDIX Есть;
 Максимальное количество открытых сокетов 64;
 Напряжение гальванической изоляции, В (АС) 1500;
 Тип разъема RJ45 (8 конт.)

1.4.4 Параметры 2G GSM-модема (только для исполнения -1GM1RS)

Рабочие частоты, МГц 850 / 900 / 1800 / 1900;
 Класс передачи данных GPRS multi-slot class 12 (↑85.6 Kbps);
 Класс мощности 4 (2 Вт в диапазонах 850 / 900 МГц);
 1 (1 Вт в диапазонах 1800 / 1900 МГц);
 Формат SIM-карты Mini-SIM (2FF);
 Размеры SIM-карты, мм 25×15×0,76;
 Тип антенного разъёма на модуле Female SMA.

1.4.5 Габаритные размеры

Габаритные размеры показаны на рисунке 1.3.

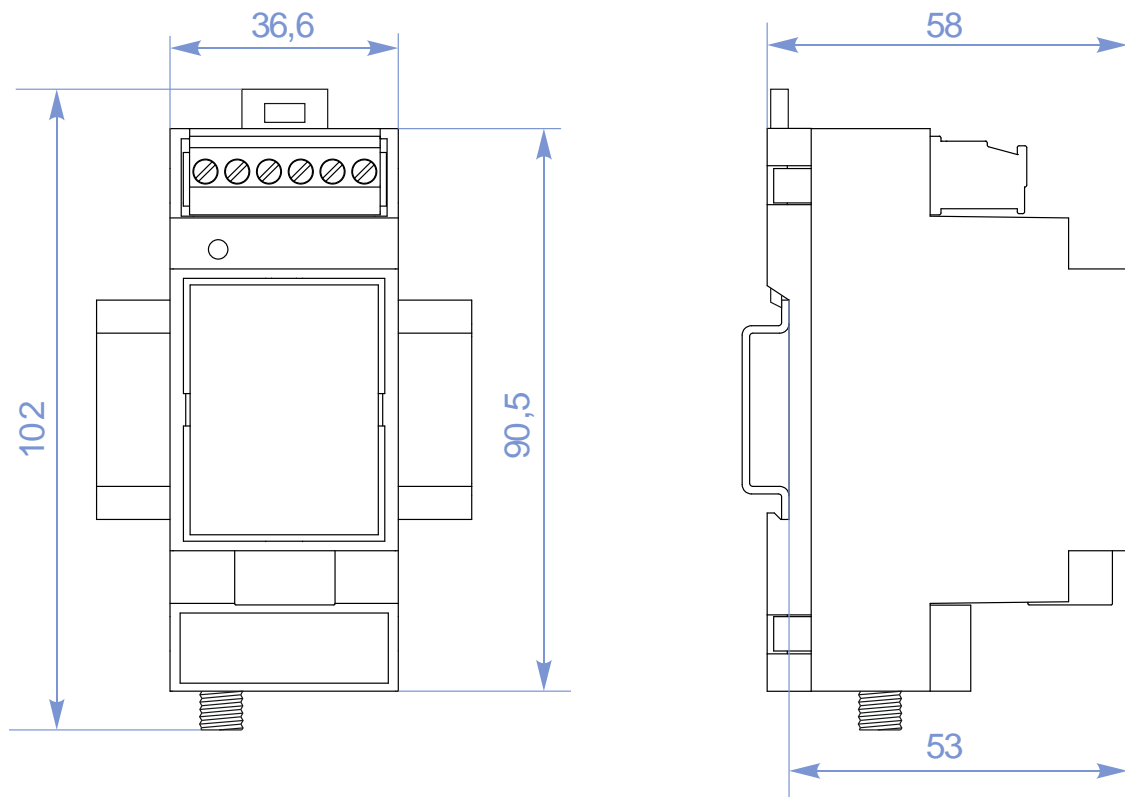


Рисунок 1.3 – Габаритные размеры

1.5 Устройство и работа

1.5.1 Общее описание конструкции модуля

Модуль PL307 выполнен в пластиковом корпусе и предназначен для монтажа на DIN-рейку. Для эксплуатации устройства на открытом воздухе – его необходимо смонтировать внутри оболочки со степенью защиты IP54 или выше.

Модуль содержит разъёмный блок клемм для подключения внешних цепей, порт Ethernet, слот для установки карты памяти microSD, а также антенный разъём и слот для установки SIM-карты (для исполнений со встроенным GSM-модемом).

Под верхней крышкой находится батарейка типа CR2032 и переключатели режимов работы SW1 и SW2. Слот SIM-карты находится под боковой крышкой. Для снятия крышек необходима тонкая шлицевая отвертка (не более 3мм). Способ снятия показан на рисунке ниже.

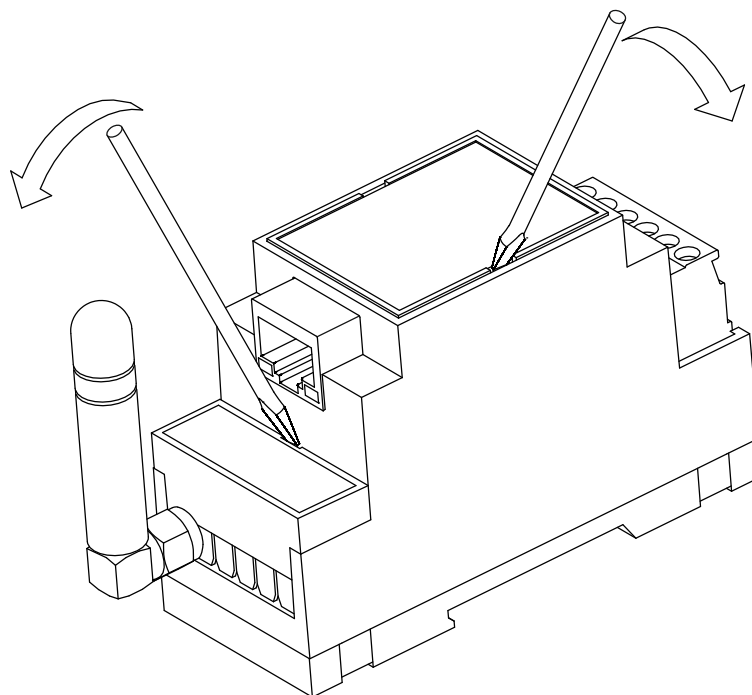


Рисунок 1.4 – Снятие крышек

На рисунке ниже показан модуль со снятыми верхней и боковой крышками.

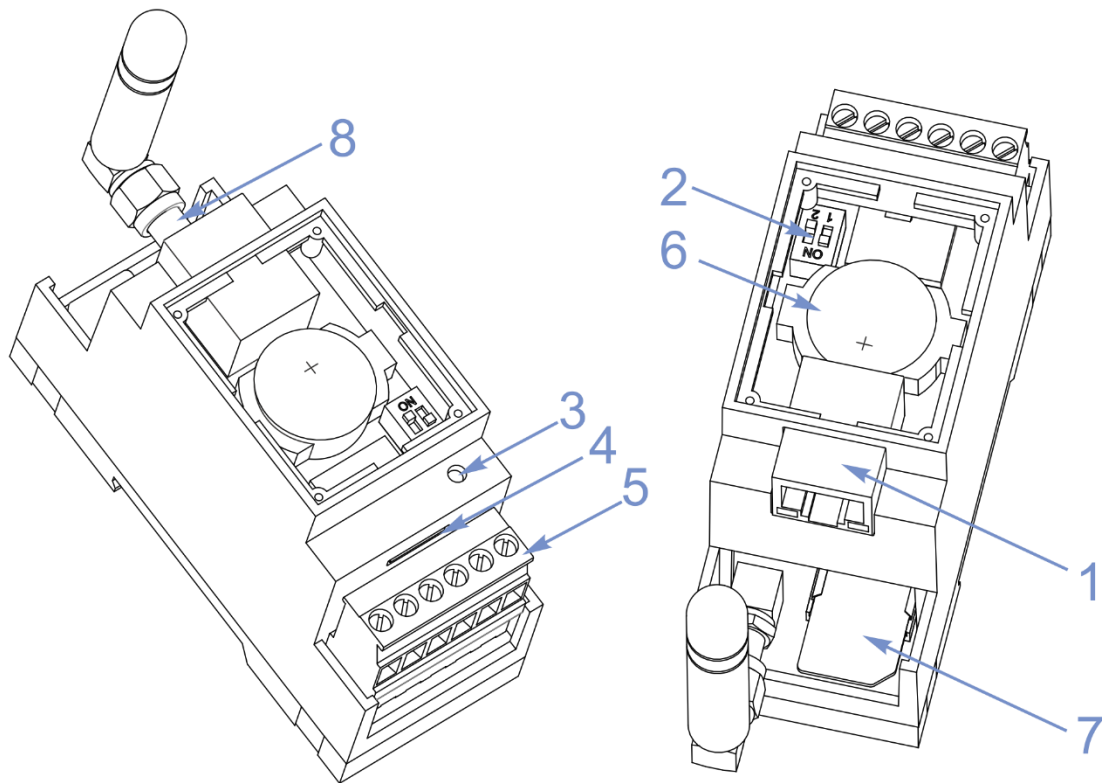


Рисунок 1.5 – Устройство модуля PL307

- 1 – разъем Ethernet;
- 2 – переключатель режимов работы SW1 и SW2;
- 3 – светодиод STATUS;
- 4 – слот microSD-карты;
- 5 – клеммный блок;
- 6 – батарея часов;
- 7 – слот SIM-карты;
- 8 – антенный разъем.

1.5.2 Монтаж модуля

Модуль PL307 предназначен для монтажа на DIN-рейку высотой 7,5 или 15 мм. Модуль фиксируется на рейке пластиковой защёлкой. Чтобы демонтировать контроллер – нужно отжать отверткой язычок защёлки.

1.5.3 Назначение клемм

Назначение клемм приведено в таблице ниже.

Таблица 1.1 – Назначение клемм

Клемма	Цепь	Описание
1	A	Положительная цепь (+) интерфейса RS-485 (PORT1)
2	B	Отрицательная цепь (-) интерфейса RS-485 (PORT1)
3	SG	Сигнальная «земля» интерфейса RS-485 (PORT1)
4	NC	Клемма не подключена
5	+U	Ввод питания «плюс»
6	-U	Ввод питания «минус»

Допускается применять только медные одножильные или многожильные провода. Применение проводов с алюминиевой жилой недопустимо.

Многожильные провода предварительно рекомендуется обжать специальным трубчатым наконечником, одножильные провода устанавливаются без обжатия наконечником. Поверхность жил проводов должна быть чистой и свободной от окислов.

Рекомендуется использовать многожильные провода сечением от 0,25 до 1,5 мм².

1.5.4 Органы индикации и управления

На лицевой части модуля на «ступеньке» расположен светодиод «STATUS», отражающий состояние модуля. Светодиоды LINK и POW расположены непосредственно на разъеме RJ-45.

Светодиод STATUS может находиться в следующих состояниях:

- не светится – контроллер обесточен или неисправен;
- светится постоянно – контроллер неисправен;
- мигает зеленым с частотой 2-3 раза в с – нормальная работа;
- мигает красным с частотой 2-3 раза в с – ошибка конфигурирования или внутренняя ошибка модуля;
- мигает попеременно красным и зеленым (приблизительно с частотой 15 Гц) – режим бутлоадера.

Светодиод LINK - горит зеленым, если есть физическое подключение к порту Ethernet.

Светодиод ACT - мигает желтым, если происходит обмен пакетами между модулем PL307 и абонентами в сети Ethernet.

Переключатели режимов работы SW1 и SW2 находятся под крышкой модуля. Переключатели опрашиваются только один раз при инициализации модуля после любого сброса (питание, WDT, команда сброса и т.д.).

Переключатель SW2 (на рисунке справа), установленный в положение ON, принудительно включает режим бутлоадера. В этом режиме модуль будет пытаться обновить прошивку с SD-карты. Файл прошивки «PL307.bfm» должен находиться на SD-карте в папке «system». В этом режиме отключены все каналы связи, на запросы модуль не отвечает, значение переключателя SW1 игнорируется. Более подробно об обновлении прошивки см. раздел «Обновление встроенного ПО».

1.5.5 Настройки связи по умолчанию

Переключатель SW1 установленный в положение ON (SW2 должен быть в положение OFF) активирует настройки связи интерфейса Ethernet в значения по умолчанию. Настройки связи устанавливаются в значения по умолчанию временно, пока включен SW1, а при отключении и пересбросе модуля применяются пользовательские настройки связи.

Параметр	В режиме конфигурирования SW1 – ВКЛ / SW2 - ОТКЛ	В новом устройстве по умолчанию
IP-адрес	192.168.0.223	Определяются автоматически (включен DHCP)
Маска подсети	255.255.254.0	
Шлюз	192.168.0.1	
DNS-сервер	77.88.8.88	
Web-сервер	Включен	Включен
Порт Web-сервера	80	80
Пароль к странице конфигурации	111111	111111
Адрес устройства PL307 на шине Modbus	247	246

Доступ к странице конфигурации: <http://192.168.0.223/cfgPL/index.html> при включенном SW1 (внимание – адрес вводить вместе с завершающим символом «/»). Когда SW1 отключен – необходимо указать реальный IP-адрес вместо 192.168.0.223.

1.5.6 Работа с настройками связи по умолчанию

Модуль PL307 с включенным переключателем SW1 функционирует обычным образом. Этот факт может быть использован для временной работы с фиксированными настройками, в ситуации, когда локальная сеть в здании ещё отсутствует.

Другими словами, можно смонтировать модуль PL307 комплектно с оборудованием, к которому он подключается (контроллеры, модули ввода/вывода, пульта и т.д.), произвести в модуле PL307 все настройки, которые должны действовать в локальной сети (когда она появится), т.е. как правило это автоопределение настроек связи. Но переключатель SW1 перевести во включенное положение.

Для работы сети Ethernet необходимо установить локальный коммутатор портов Ethernet. Все устройства, работающие с PL307 через Ethernet, должны будут подключаться к коммутатору. При этом модуль PL307 будет работать штатно, но с настройками по умолчанию. Получаем автономную локальную сеть.

После того, как локальная сеть здания будет готова, коммутатор соединяется с этой локальной сетью, а на модуле PL307 переключатель SW1 отключается (этот может сделать местный обслуживающий персонал, что-либо менять в настройках уже не требуется). После сброса питания будут приняты настройки из конфигурации и PL307 сможет работать в локальной сети здания с ранее сделанными настройками.

1.5.7 Батарея

Батарея используется для питания часов реального времени, когда модуль обесточен. Используется литиевая батарея типа CR2032.

Модуль не имеет каких-либо средств контроля степени заряженности батареи, и то, что она разрядилась можно контролировать только по косвенным признакам (сброс часов при отключенном питании).

Расчётный срок времени работы часов от свежей батареи составляет порядка 2 лет (когда модуль PL307 не запитан). Когда на модуль подано питание – энергия батареи не тратится. Срок службы (сохраняемости) для литиевых батарей составляет, как правило, порядка 10 лет. Для ответственных применений рекомендуется выполнять плановую замену батареи 1 раз в 3-5 лет.

Замену батареи можно выполнить на работающем модуле.

2 Настройка режимов работы

2.1 Установка связи с модулем

2.1.1 Общие сведения

Настройка модуля осуществляется при помощи браузера. Для этого необходимо установить соединение через порт Ethernet. Как правило, необходимо наличие локальной сети предприятия/дома/квартиры, либо такая сеть должна быть специально организована для соединения компьютера и устройства. Также модуль PL307 обладает поддержкой HP Auto-MDIX (автоматическое переназначение пар RX/TX), что позволяет выполнить прямое подключение точка-точка между модулем и компьютером обычным патч-кордом.

По умолчанию IP-адрес модуля 192.168.0.223, маска подсети 255.255.254.0. И он может работать в сети, имеющей IP-адреса вида 192.168.0.xxx или 192.168.1.xxx. Ниже будут рассмотрены разные способы подключения.

Чтобы открыть страницу настроек необходимо в браузере набрать путь следующего вида «<http://192.168.0.223/cfgPL/index.html>». Здесь указан IP адрес по умолчанию для нового модуля, если он был изменен, то необходимо указать тот, что прописан в настройках модуля. Далее будет выдано приглашение для ввода пароля для доступа к странице настроек. Для нового модуля и для модуля с включенным переключателем SW1 для страницы настроек установлен пароль по умолчанию 111111. После ввод пароля открывается страница настроек модуля.

2.1.2 Варианты подключения

Вариант 1. Имеется локальная сеть предприятия или дома/квартиры с IP-адресами вида 192.168.0.xxx или 192.168.1.xxx и маской подсети 255.255.255.0 (такие настройки имеют по умолчанию большинство обычных роутеров). В этом случае нужно просто подключить модуль PL307 в свободный порт коммутатора, предварительно убедившись, что IP-адрес 192.168.0.223 не занят. Далее можно устанавливать соединение с модулем с любого устройства в локальной сети. Рекомендуется в настройках роутера (в сети предприятия или дома) зарезервировать IP-адрес, назначенный модулю PL307, чтобы DHCP-сервер роутера никому не присваивал этот адрес. Также, если в роутере зарезервировать IP-адрес за модулем PL307, то в модуле можно активировать автоматическое получение настроек и все параметры связи IPv4 будут определяться модулем автоматически.

Вариант 2. Имеется локальная сеть с IP-адресами отличными от 192.168.0.xxx и 192.168.1.xxx (например 192.168.8.xxx). В этом случае можно временно (или навсегда, если это не принципиально) изменить настройки роутера, чтобы IP-адреса соответствовали виду 192.168.0.xxx, далее подключиться к модулю PL307 с любого устройства в локальной сети и изменить его IP-адрес, чтобы он соответствовал вашей подсети, после чего настройки роутера можно вернуть обратно.

Вариант 3. Нет локальной сети в данный момент, либо сеть есть, но её настройки не подходят и поменять их нельзя. В этом случае можно либо попытаться установить прямое соединение «точка-точка» между компьютером и модулем PL307 обычным патч-кордом (не все сетевые адаптеры это позволяют сделать), либо использовать вспомогательный коммутатор или хаб. В компьютере в настройках протокола TCP/IP необходимо вручную прописать его IP-адрес, например 192.168.0.2, и маску подсети 255.255.255.0. Далее можно установить соединение с модулем PL307 и установить его параметры связи в нужное значение.

При подключении соединительного кабеля (патч-корда) к разъему RJ45 модуля необходимо убедиться, что светодиод LINK светится зеленым. Если светодиод погашен, то подключения физически нет и дальнейшие действия не имеют смысла. Возможные причины отсутствия подключения: поврежден кабель, неправильно обжат разъем, обесточен коммутатор, или при прямом подключении «точка-точка» сетевым адаптером компьютера это соединение не поддерживается.

Далее, рекомендуется проверить «пинг» до модуля, для этого необходимо запустить командный интерпретатор Windows. Запустить его можно, например, так: нажать «Пуск/Выполнить» и в строке набрать «cmd.exe» (без кавычек), либо найти и запустить в файловом менеджере «C:\Windows\System32\cmd.exe». В открывшемся окне набрать следующую команду: `ping xxx.xxx.xxx.xxx`, где вместо x вписать настроенный IP-адрес модуля (по умолчанию 192.168.0.223). Если команда не выполняется: нет ответа, утеряны пакеты, общий сбой, необходимо перевести переключатель SW1 в положение ON и пересбросить питание модуля, тем самым установив параметры связи в значения по умолчанию. Выполнение команды (есть ответы и ни одного пакета не потеряно) свидетельствует об успешном подключении к сети Ethernet, после чего можно приступить непосредственно к настройке.

2.2 Настройка режима работы

На странице «Основные настройки» настраивается:

режим работы модуля;

собственный адрес модуля на шине Modbus TCP;

ответ модуля запрашиваемому устройству в режиме «Шлюз», когда внешнее устройство не отвечает на запрос.

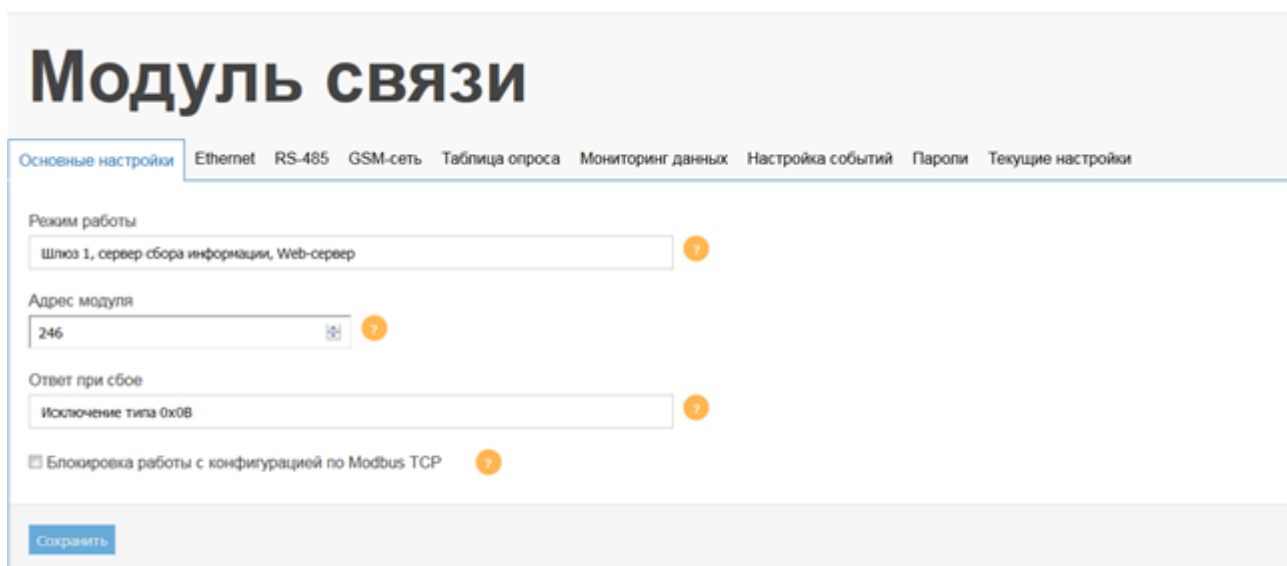


Рисунок 2.1 – Страница основных настроек модуля PL307

Возможные варианты ответа модуля PL307 запрашивающему устройству (Modbus TCP), если опрашиваемое устройство (Modbus RTU) не отвечает:

«Исключение Modbus 0x0B» (Gateway target device failed to respond) – ответ, предусмотренный стандартом Modbus;

«Старое значение» - запрашиваемому устройству возвращается предыдущее сохраненное значение, данный вариант может быть полезен, если запрашиваемое устройство не может корректно обрабатывать исключения.

Модуль PL307 может работать в следующих режимах (см. Рисунок 2.1):

- Шлюз 1: Modbus TCP → Modbus RTU;
- Шлюз 2: Modbus RTU → Modbus TCP;
- Шлюз 3: Modbus TCP → Modbus TCP;
- Шлюз 1, сервер сбора информации, Web-сервер;
- Шлюз 3, сервер сбора информации, Web-сервер;
- Технологическая программа (*для специальных применений*).

Режимы работы рассмотрены ниже.

Режим «Технологическая программа» не реализован в типовой прошивке, которой по умолчанию прошиваются модули, и предназначен для специализированных применений модуля. В этом режиме выполняется встроенная технологическая программа, которая работает с данными, полученными сервером сбора информации. Технологическая программа имеет свой набор регистров Modbus, которые доступны для чтения и записи. В этом режиме также работает встроенный WEB-сервер (набор данных, доступных через WEB-сервер, описан на вкладках соответствующих технологических программ).

Соответствующая технологическая программа может быть разработана по заявке.

2.2.1 Режим «Шлюз 1: Modbus TCP → Modbus RTU»

В этом режиме (будем называть его кратко «Шлюз 1») модуль является аппаратным конвертером протокола Modbus TCP в Modbus RTU. Режим предназначен для опроса ведомых устройств, работающих по RS-485, мастер-устройствами с Ethernet. Модуль принимает запросы по протоколу Modbus TCP и транслирует их в запросы Modbus RTU, а ответы ведомых устройств в обратном направлении. Во избежание конфликтов и потерь пакетов в модуле реализована очередь. Это позволяет одновременно опрашивать внешние модули сразу несколькими различными абонентами. Никаких дополнительных настроек, кроме настроек связи модулю в этом режиме не требуется.

В режиме работы «Шлюз 1», такие функции как архивирование данных и web-сервер работать не могут.

Режим «Шлюз 1» может работать и когда модуль является сервером (принимает запросы на TCP-соединения) и когда модуль сам устанавливает TCP-соединение с удаленным сервером, который затем ведёт опрос по протоколу Modbus TCP. Эти настройки делаются на странице «Ethernet».

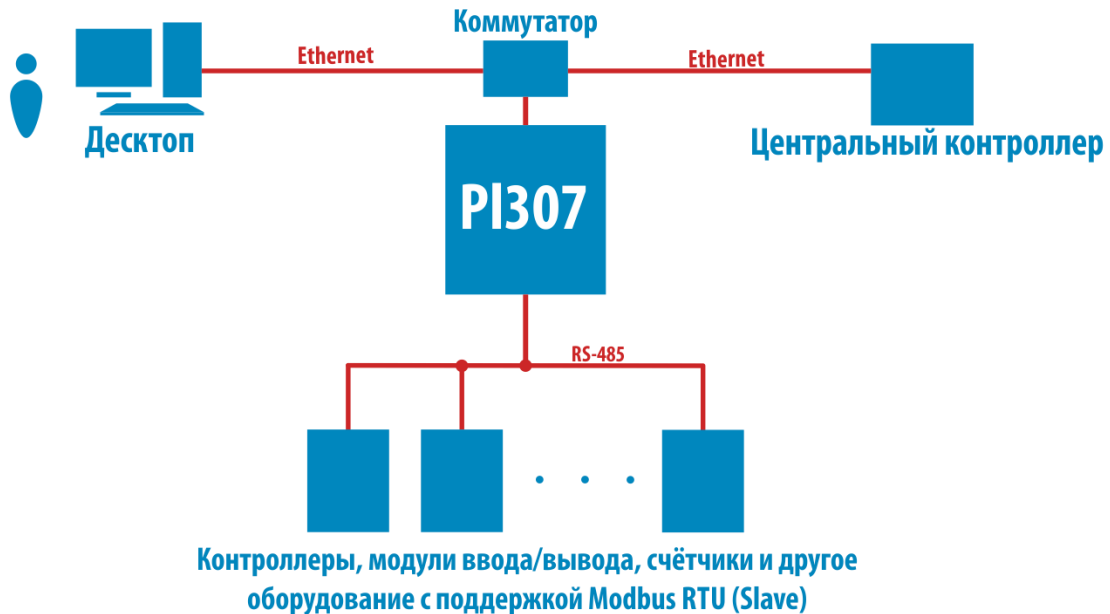


Рисунок 2.2 – Схема работы модуля PL307 в режиме «Шлюз»

Если в запросе Modbus TCP будет указан собственный адрес модуля PL307, то в ответ возвращаются данные самого модуля, а не внешнего устройства (запрос на шине RS-485 не формируется).

2.2.2 Режим «Шлюз 2: Modbus RTU → Modbus TCP»

Режим «Шлюз 2: Modbus RTU → Modbus TCP» (далее «Шлюз 2») предназначен для преобразования запросов с Modbus RTU (шина RS-485) в запросы Modbus TCP (Ethernet). На шине RS-485 должно находиться мастер-устройство, а на шине Ethernet соответственно ведомое. В этом режиме работы транслируются все запросы в обоих направлениях. Настройки TCP-соединения ведомого устройства задаются в разделе Ethernet/Socket1.

Работа по Ethernet идёт через Socket1, поэтому он должен быть обязательно настроен как Modbus TCP Master.

Со стороны RS-485 может находиться только одно мастер-устройство, а также могут находиться ведомые устройства, которые опрашиваются тем же мастером. При этом следует иметь в виду, что запросы к этим ведомым устройствам будут также транслироваться модулем PL307 в запросы Modbus TCP на шину Ethernet, поэтому не должно быть пересекающихся адресов Modbus.

Если в запросе Modbus RTU будет указан собственный адрес модуля PL307, то в ответ возвращаются данные самого модуля, а не внешнего устройства (запрос на шине Ethernet не формируется).

2.2.3 Шлюз 3: Modbus TCP → Modbus TCP

Шлюз Modbus по сети Ethernet (протокол Modbus TCP). Мастер-устройство устанавливает соединение по Ethernet или GSM через "Сервер Modbus TCP Slave" или "TCP / Socket 2" в режиме "Modbus TCP Slave", а ведомые устройства подключаются, также, по Ethernet через "TCP / Socket 1" в режиме "Modbus TCP Master". В этом режиме работы все транзитные запросы транслируются на "TCP / Socket 1". Настройки TCP-соединения ведомых устройств задаются в разделе "Ethernet / Socket 1". Необходимое

условие работы - TCP Socket 1 должен быть включен и настроен, как "Modbus TCP Master";

2.2.4 Режим «Шлюз 1, сервер сбора информации, Web-сервер»

В этом режиме (далее «Сервер сбора данных») модуль, дополнительно к функциональности режима «Шлюз 1», ведет непрерывный опрос устройств по протоколам Modbus RTU и Modbus TCP, поддерживая в своей памяти актуальную базу текущих значений параметров. При запросе каких-либо данных со стороны Master-устройств – они тут же извлекаются из локальной базы, минуя запрос-ответ от внешнего модуля.

Этот режим позволяет значительно ускорить опрос медленных Slave-устройств на шине RS-485, со стороны Master-устройств, а также режим работы «Сервер сбора данных» является необходимым условием для работы таких функций как архивирование данных и Web-сервер.

Переменные, которые необходимо опрашивать, описываются на странице настроек «Таблица опроса модулей». Эта таблица хранится на SD-карте и может быть также туда записана в виде файла. Каждой переменной присваивается символьное имя, которое в дальнейшем используется для обращения к текущим или архивным данным этой переменной при работе web-сервера.

При получении запроса чтения данных Modbus TCP модуль сначала проверяет описаны ли запрашиваемые данные в таблице опроса. Если будут запрашиваться неописанные переменные или будет описана только часть переменных из запроса, то модуль PL307 будет генерировать запрос по протоколу Modbus RTU, аналогично работе в режиме «Шлюз 1».

При получении модулем запроса Modbus TCP на запись данных работа происходит аналогично работе в режиме «Шлюз 1». То есть формируется и отправляется запрос Modbus RTU на шину RS-485, а полученный ответ преобразуется и отправляется в формате Modbus TCP.

Режим «Сервер сбора данных» может работать и когда модуль является сервером (принимает запросы на TCP-соединения), а также когда модуль сам устанавливает TCP-соединение с удаленным сервером, который затем ведёт опрос по протоколу Modbus TCP. Эти настройки делаются на странице «Ethernet».

2.2.5 Шлюз 3, сервер сбора информации, Web-сервер

Режим аналогичен предыдущему (см. п. 2.2.4), но Slave-устройства подключаются через интерфейс Ethernet по протоколу Modbus TCP.

2.3 Настройка параметров связи порта Ethernet

2.3.1 Настройки типа сети и IPv4

Данная настройка выполняется со страницы настроек «Ethernet» (см. рисунок ниже) подраздел «Настройки Ethernet» и «Настройки IPv4».

Параметр «Тип сети Ethernet» позволяет принудительно указать тип сети, либо позволить модулю сделать это автоматически (Auto-Negotiation). Если установлен переключатель SW1 в положение ON, то данное поле игнорируется, и модуль определяет тип сети автоматически.

Модуль связи

Основные настройки **Ethernet** RS-485 GSM-сеть Таблица опроса Мониторинг данных Настройка событий Пароли Текущие настройки

Настройки Ethernet

Тип сети Ethernet
 ?

Настройки IPv4

Получить автоматически ?

IP-адрес
 ?

Маска подсети
 ?

IP шлюза
 ?

IP DNS-сервера
 ?

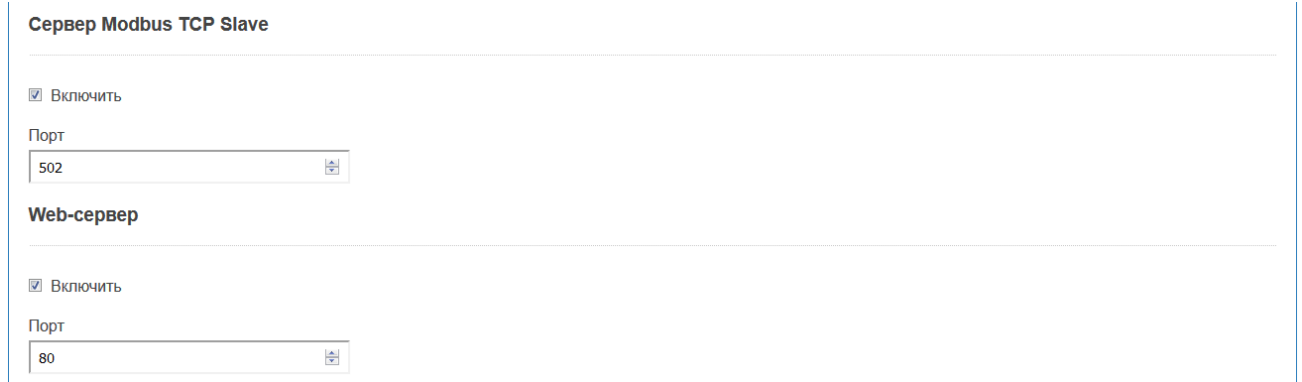
Рисунок 2.3 – Страница настроек Ethernet

Настройки связи по протоколу TCP/IPv4 можно получить автоматически, если установить соответствующую галочку в блоке «Настройки IPv4» или задать вручную. После редактирования настроек необходимо нажать кнопку «Сохранить».

Вариант с автоматическими настройками безусловно подходит для режима связи с внешним сервером сбора информации, т.е. когда модуль PL307 является инициатором соединения и его текущий IP-адрес может быть любым. Однако обратите внимание, что если модуль PL307 планируется использовать в режиме сервера, т.е. когда он сам принимает входящие соединения, то IP-адрес модуля должен быть постоянным и не изменяться. В этом случае автоматические настройки можно использовать только если закрепить за модулем статический IP-адрес в роутере (устройстве, содержащем DHCP-сервер), как правило с привязкой к MAC-адресу.

2.3.2 Настройки режимов сервера

Возможность модуля принимать входящие соединения по определенным протоколам активируется соответственно в подразделах настроек «Сервер Modbus TCP Slave» и «Web-сервер» (см. Рисунок 2.4). Необходимо установить соответствующую галочку и указать порт, который модуль должен прослушивать.



The screenshot shows a configuration interface for a server. It is divided into two sections: "Сервер Modbus TCP Slave" and "Web-сервер".

- Сервер Modbus TCP Slave:** Includes a checked checkbox "Включить" (Enable), a "Порт" (Port) dropdown menu set to "502".
- Web-сервер:** Includes a checked checkbox "Включить" (Enable), a "Порт" (Port) dropdown menu set to "80".

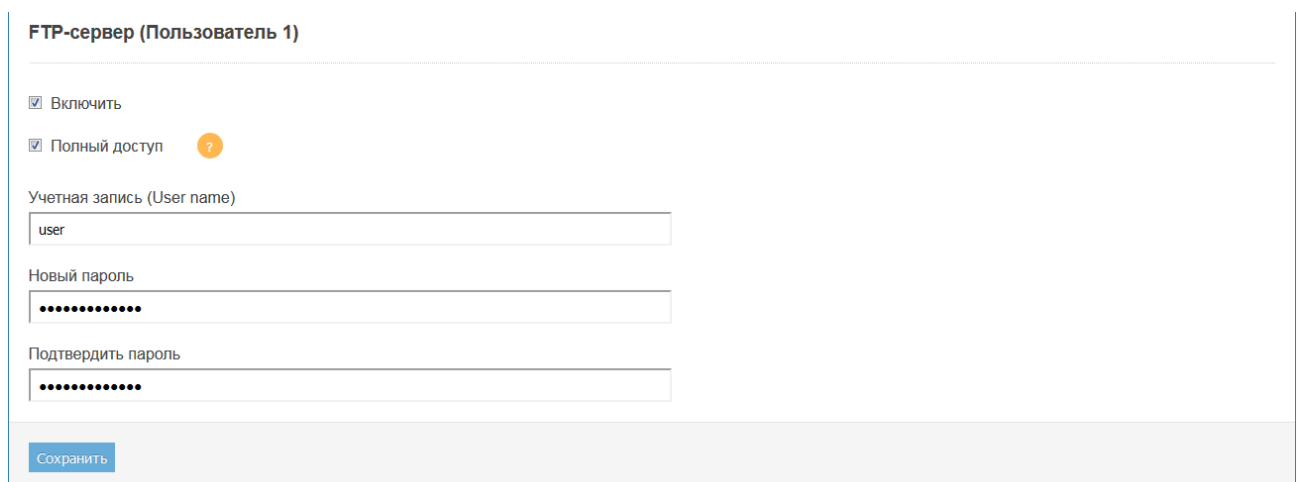
Рисунок 2.4 –Настройки режимов сервера

Для обработки входящих http-запросов нужно включить «Web-сервер» и также настроить номер порта. Если Web-сервер не нужен, его можно отключить, убрав галочку «Включить», однако страницы конфигурации без включенного Web-сервера также будут недоступны. В этом случае, если необходимо изменить какие-либо параметры конфигурации, то необходимо использовать переключатель SW1. Отключенный web-сервер гарантирует отсутствие возможности изменить несанкционированно настройки конфигурации. Для работы пользовательского Web-сервера его необходимо создать и загрузить на карту памяти microSD. Как это сделать описывается ниже.

2.3.3 FTP-сервер

Для работы с файлами можно настроить доступ для двух пользователей. См. Рисунок 2.5. Галочка «Полный доступ» открывает полный доступ по FTP доступ ко всем файлам на SD-карте. Если данная опция не включена, то по FTP будут доступны только файлы из директории «archive».

Для соединений должен использоваться пассивный режим (по умолчанию в FTP-клиентах).



The screenshot shows the "FTP-сервер (Пользователь 1)" configuration screen. It includes the following elements:

- Checked checkboxes for "Включить" (Enable) and "Полный доступ" (Full access), with a question mark icon next to the latter.
- Input field for "Учетная запись (User name)" containing the text "user".
- Input field for "Новый пароль" (New password) with masked characters (dots).
- Input field for "Подтвердить пароль" (Confirm password) with masked characters (dots).
- A blue "Сохранить" (Save) button at the bottom.

Рисунок 2.5 –Настройки FTP

2.3.4 Настройки исходящих TCP-соединений

Модуль PL307 может как принимать входящие соединения через порт Ethernet, так и инициировать исходящие соединения с внешними устройствами. При этом вышеописанные режимы «Шлюз» и «Сервер сбора данных» могут работать для обоих способов соединения. Список настроек приведен на рисунке ниже.

Для исходящих соединений может быть настроено два TCP-сокета. При этом для каждого исходящего сокета должен быть выбран определенный режим работы (параметр «Тип протокола связи»).

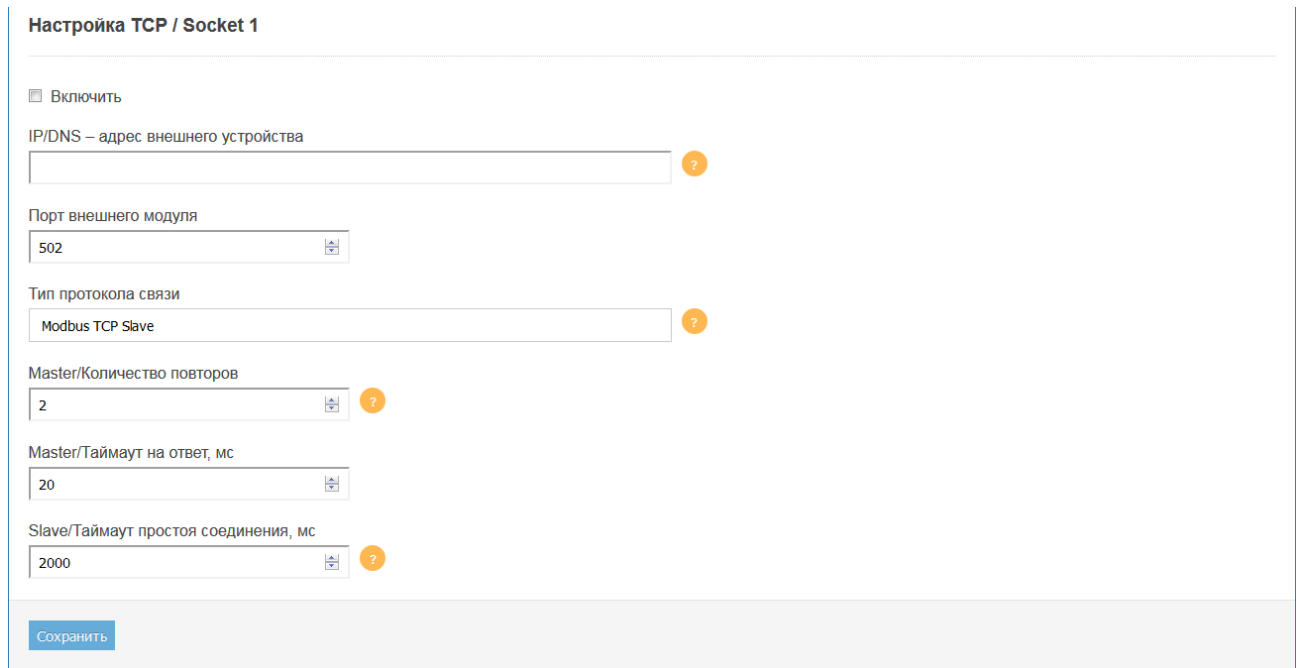


Рисунок 2.6 –Настройки исходящих TCP-соединений

Могут быть настроены следующие режимы работы (протоколы связи):

«Modbus TCP Slave» – Модуль PL307 устанавливает TCP-соединение с внешним устройством (сервером сбора информации, система удаленного мониторинга и т.д.), которое может находиться в глобальной сети Интернет. Далее внешнее устройство ведет чтение/запись данных по протоколу Modbus TCP. Данный режим удобен тем, что модулю PL307 нет необходимости иметь статический IP-адрес.

«Modbus TCP Master» – Модуль PL307 устанавливает TCP-соединение с внешним устройством и далее ведет чтение/запись Modbus регистров этого устройства согласно описанных переменных на вкладке «Таблица опроса».

Для настройки соединения необходимо указать IP-адрес (или DNS-имя) и входящий порт внешнего устройства, а также протокол, по которому этот сокет будет работать (см. выше).

Для режима «Modbus TCP Master» настраивается ещё два параметра: «Master/Количество повторов» и «Master/Таймаут на ответ, мс». Эти параметры соответственно устанавливают количество повторных запросов, если не получен ответ от внешнего устройства, а также время ожидания ответа.

Для режима «Slave/Таймаут простоя соединения, мс» настраивается время, в течение которого должен прийти хотя бы один запрос от внешнего устройства. Если в

течение указанного времени запрос не приходит, то TCP-соединение закрывается и открывается заново. Время задаётся в мс от 100 до ($2^{32}-1$).

2.4 Настройка RS-485

Страница настроек порта RS-485 показана на рисунке ниже.

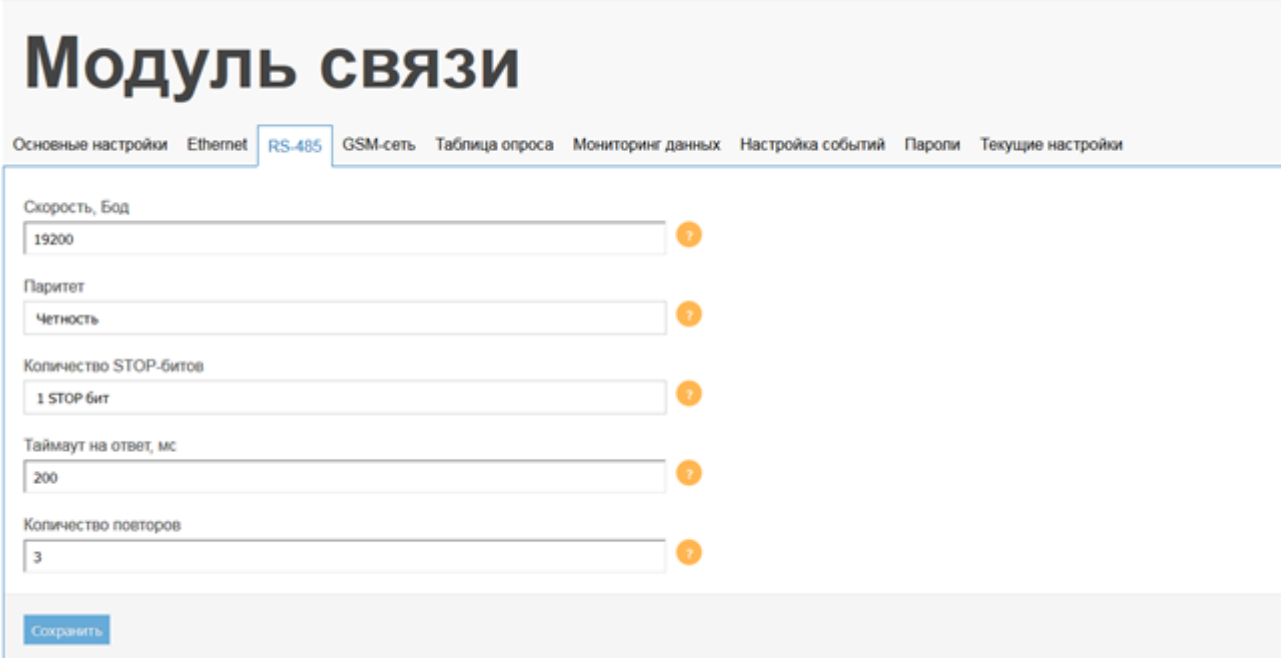


Рисунок 2.7 – Страница настроек RS-485

Внимание! После настроек RS-485 модуль необходимо перезагрузить.

2.5 GSM-сеть

Настройки соединения через встроенный GSM-модем показаны на рисунке ниже.

Параметр «Использовать GSM-сеть как основную» направляет все служебные пакеты в GSM-сеть. Также исходящие TCP-соединения будут работать через сеть GSM, если она находится в рабочем состоянии.

Модуль связи

Основные настройки Ethernet RS-485 **GSM-сеть** Таблица опроса Мониторинг данных Настройка событий Пароли Текущие настройки

Настройки GSM-сети

Включить GSM-сеть

Использовать GSM-сеть, как основную. ?

Настройки APN

APN-имя GSM-сети

?

Использовать пароль для APN-сети

Логин для APN-сети

?

Пароль для APN-сети

Подтвердить пароль

Рисунок 2.8 – Настройки GSM-сети

2.6 Таблица опроса

В таблице опроса описывается перечень внешних устройств и перечень переменных, которые должны в автоматическом режиме считываться из этих устройств. Значения переменных, полученные из внешних устройств, хранятся в оперативной памяти модуля PL307. Эти данные могут быть незамедлительно предоставлены при запросе по Modbus TCP или HTTP (при работе встроенного сайта).

На рисунке ниже показаны настройки временных параметров.

Модуль связи

Основные настройки Ethernet RS-485 GSM-сеть **Таблица опроса** Мониторинг данных Настройка событий Пароли Текущие настройки

Настройки опроса

Синхронизировать каналы связи опроса переменных ?

Пауза перед началом опроса переменных после перезагрузки модуля, с
 ?

Период опроса, мс
 ?

Пауза между пакетами опроса переменных по RS-485/Port1, мс

Пауза между пакетами опроса переменных по TCP/Socket1, мс

Пауза между пакетами опроса переменных по TCP/Socket2, мс

Рисунок 2.9 – Настройки опроса переменных

«Синхронизировать каналы связи опроса переменных» - если отключено, то опрос данных по каждому каналу связи работает асинхронно, при этом период опроса контролируется по каждому каналу индивидуально. Если синхронизация включена, то опрос данных, также, ведётся параллельно, но по завершению опроса, новый цикл опроса начинается только когда будет завершён опрос по каждому из каналов связи. Включение синхронизации может быть целесообразно для сохранения целостности данных и корректного выполнения скриптов.

«Пауза перед началом опроса переменных после перезагрузки модуля» - нужна для опроса модулей, с долгой инициализацией после включения питания для исключения ошибок чтения.

«Период опроса» - Задаёт периодичность, с которой выполняется опрос внешних устройств. Если заданный период окажется меньше времени реального опроса, то опрос будет вестись с фактическим минимально-возможным временем.

«Пауза между пакетами опроса переменных по RS-485/Port1» (TCP/Socket1, TCP/Socket2) – может быть полезна для уменьшения нагрузки на внешние устройства.

Для настройки опроса внешних устройств сначала нужно описать таблицу внешних устройств. Пример таблицы приведен на рисунке ниже.

Таблица устройств

Имя устройства	Тип канала связи	Адрес устройства	Формат данных	Комментарии	Счетчики пакетов		Статус связи	
					Успешных	Ошибочных		
JL206_1	RS485/PORT1	1	b1,b0,b3,b2	Основной контроллер	318703	19128	●	Удалить
доп_контр	RS485/PORT1	2	b1,b0,b3,b2	Вспомогательный контроллер	0	58812	●	Удалить

[Добавить](#)

Рисунок 2.10 – Настройка списка устройств

В таблице для каждого устройства описываются следующие параметры:

- имя устройства – используется далее в таблице переменных, строка в формате UTF-8; строка должна состоять из цифр и русских и/или английских букв, и она не должна быть длиннее 32 символов; имена устройств не должны повторяться;
- канал связи, к которому подключено внешнее устройство; кроме физического канала связи можно указать «Local» для описания локальных переменных;
- адрес устройства на шине Modbus - должен находиться строго в диапазоне от 1 до 255; значения 0 и больше 255 - вводить запрещено;
- формат данных – указывает на чередование байт в 16- и 32-битных словах; формат данных выбирается для всего устройства и его нельзя изменить для одной конкретной переменной
- комментарий – произвольная текстовая строка.

Кроме этого, приводится текущая статистика успешных и ошибочных запросов к внешнему устройству и текущий статус связи в виде зелёного (есть связь) или красного (нет связи) кружка.

Кнопками «Добавить» и «Удалить» можно соответственно добавлять или удалять строки из таблицы.

Ниже, на рисунке приведен пример таблицы опроса переменных.

Таблица опроса

Имя устройства	Тип регистра	Индекс регистра	Тип данных	Размер данных	Множитель данных	Символьный тэг данных	Вкл./Откл. архивации	Текущее значение	
JL206_1	Input	10	UINT16	1	1	cur_mode	Включена	319(OK)	Удалить
JL206_1	Input	11	UINT32	2	1	status	Включена	0(OK)	Удалить
JL206_1	Input	13	SINT16	1	10	cur_t	Включена	0.0(OK)	Удалить
JL206_1	Input	14	UINT16	1	1	cur_t_sts	Включена	12(OK)	Удалить
JL206_1	Input	15	UINT16	1	10	cur_h	Включена	0.0(OK)	Удалить
JL206_1	Input	16	UINT16	1	1	cur_h_sts	Включена	5(OK)	Удалить
JL206_1	Input	17	UINT16	1	100	cur_f	Включена	0.18(OK)	Удалить
JL206_1	Input	18	UINT16	1	1	cur_f_sts	Включена	0(OK)	Удалить
JL206_1	Input	19	UINT32	2	1	errors	Включена	264569700(OK)	Удалить
JL206_1	Input	21	UINT32	2	1	warnings	Включена	114167439(OK)	Удалить

Рисунок 2.11 – Настройка списка устройств

В таблице для каждой переменной необходимо указать параметры согласно таблице ниже. Кроме этого, можно формировать таблицу опроса через файл «fileio.csv», который располагается в папке «\sys\serv\».

Таблица 2.1 - Соответствие полей таблицы опроса и столбцов файла

Поле/ столбец	Таблица опроса	Файл fileio.csv	Описание
Имя устройства/A	Имя	Имя	Имя устройства, описанное в таблице устройств
Тип регистра/B	Input	0	Тип регистров Modbus, в которых хранится читаемая переменная
	Holding	1	
	Discrete Input	2	
	Coil	3	
Индекс регистра/C	0-65535	0-65535	Индекс первого регистра holding или input, в котором и последующих (если размер переменной больше 1 регистра) хранится переменная
Тип данных/D	UINT8L	0	Младшие 8 бит регистра без знака
	UINT8H	1	Старшие 8 бит регистра без знака
	UINT16	2	16-ти битное число без знака
	SINT16	3	16-ти битное число со знаком
	UINT32	4	32-х битное число без знака
	SINT32	5	32-х битное число со знаком
	FLOAT	6	32 битное число в формате float
	ARRAY	7	Массив 16-ти битных значений без знака
STRING	8	Строка символов	
Размер/E	1-120	1-120	Размер переменной, выраженный в кол-ве регистров. В таблице размер устанавливается автоматически для всех типов, кроме ARRAY и STRING. Их размер может быть произвольным в диапазоне от 1 до 120. Для строки размер всегда должен быть четным, и ее длина задается не в регистрах, а в количестве символов. В файле размер должен соответствовать типу переменной: UINT8L, H, UINT16, SINT16 – 1 регистр; UINT32, SINT32, FLOAT – 2 регистра.

Поле/ столбец	Таблица опроса	Файл fileio.csv	Описание
Множитель/F	1-10000	1-10000	Множитель переменной. Чтобы избежать работы с форматом float, некоторые устройства работают с целыми числами и оперируют переменными, умноженными на какое-либо число. Например, число 24,535 передается в формате 24535. Т.е. оно умножено на 1000. Если переменная выдается в таком формате, то в данном поле пишется множитель, на который она умножена или 1, если переменная имеет правильное значение изначально.
Символьный тэг данных/G	строка	строка	Текстовая строка длиной не больше 32 символа. Строки для разных переменных не должны повторяться. Запрещено использовать специальные символы, русские буквы и т.п.
Вкл./Откл. Архивации/H	Отключена	0	Архивация переменной
	Включена	1	

2.7 Мониторинг данных

2.7.1 Ведение архивов данных

На странице «Мониторинг данных» настраивается периодичность архивации, Настройки оповещения по E-mail и настройки NTP-сервера. См. Рисунок 2.12.

В архив могут сохраняться все переменные, объявленные в Таблице опроса, кроме массивов и строк. Для этого необходимо, чтобы был включен режим работы модуля «Шлюз 1, сервер сбора информации, Web-сервер» или «Шлюз 3, сервер сбора информации, Web-сервер», вставлена SD-карта и у переменной, которую необходимо архивировать, в столбце «Вкл./Откл. архивации» был выбран пункт «Включена».

При архивации на карту памяти microSD сохраняются срезы данных с периодичностью, заданной в подразделе «Настройки архивации». Период архивации задается в диапазоне от 1 до 65535 секунд.

Внимание! Если в часах реального времени не были установлены дата и время, то архивация переменных выполняться не будет!

Модуль связи

Основные настройки Ethernet RS-485 GSM-сеть Таблица опроса **Мониторинг данных** Настройка событий Пароли Текущие настройки

Настройки архивации

Период архивации, с

1 ?

Максимальный размер файла архива, МБ

20 ?

Сохранить

Рисунок 2.12 – Страница настроек мониторинга данных

На рисунке ниже приведён вид на содержимое папки с архивными данными из клиента FileZilla.

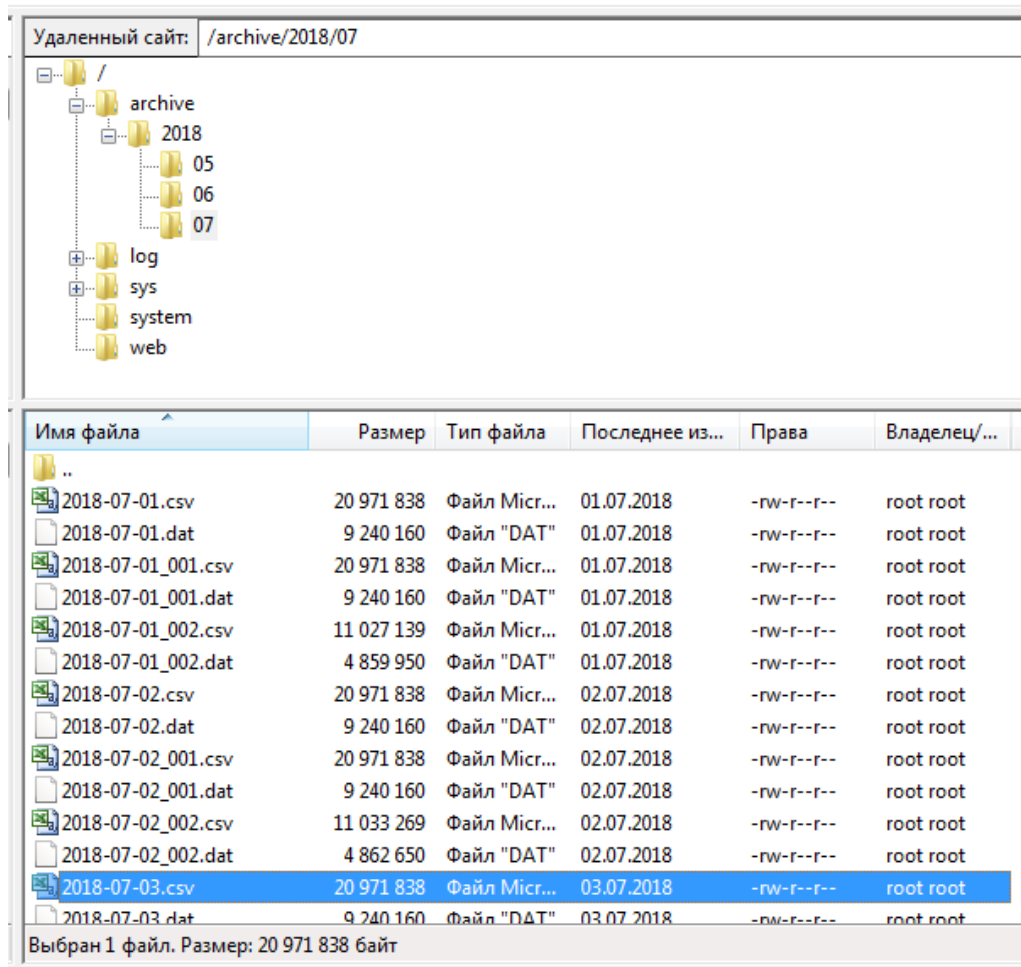


Рисунок 2.13 – Содержимое папки с архивными данными

Файлы архива хранятся в папках со строгой иерархией. Корневая папка журнала называется archive и расположена в корне SD-карты. В папке archive лежат папки с названием годов, т.е. 2017, 2018 и т.д.. В папках годов лежат папки месяцев 01, 02 ... 12. В папках месяцев лежат файлы архивов за текущий месяц. Наименование файлов имеет вид 2017-06-29_005.csv. Т.е. YYYY_MM_DD_NNN.csv. Здесь _NNN это порядковый номер файла журнала в пределах одних суток. Для самого первого файла _NNN опускается, т.е. будет выглядеть так 2017-06-29.csv, а следующим после него будет 2017-06-29_001.csv. Помимо этого, модуль параллельно ведёт запись в файл – с расширением «dat». Файлы csv могут быть открыты для анализа в программе Microsoft Excel или другой, поддерживающей данный формат. Файлы dat служебные и необходимы для работы web-сервера при запросе архивных данных из браузера.

Параметром «Максимальный размер файла архива» задаётся максимальный размер файла. При достижении этого размера или количества строк в файле 65536 - создаётся новый файл.

Формат архивного файла следующий:

	A	B	C	D	E	F	G	H	I	J	K
1	Time	cur_mode	status	cur_t	cur_t_sts	cur_h	cur_h_sts	cur_f	cur_f_sts	errors	warnings
2	12:18:01	361	0	0	9	0	5	0,18	0	264766308	102567532
3	12:18:02	361	0	0	9	0	5	0,18	0	264635236	102370928
4	12:18:03	361	0	0	10	0	5	0,18	0	264700772	102764141
5	12:18:04	361	0	0	10	0	5	0,18	0	264504164	102567534
6	12:18:05	360	0	0	9	0	5	0,18	0	264700772	102436464
7	12:18:06	360	0	0	9	0	5	0,18	0	264504164	102633072
8	12:18:07	360	0	0	9	0	5	0,18	0	264700772	102764144
9	12:18:08	360	0	0	9	0	5	0,18	0	264700772	102633072
10	12:18:09	360	0	0	9	0	5	0,18	0	264438628	102567531

Рисунок 2.14 – Вид архива в программе Excel

Первая строка – имена переменных. Вторая и последующие строки – значения переменных на данный срез времени.

Первый столбец - время с точностью до секунды. Каждый следующий столбец – это одна переменная со значениями.

Если достоверное значение переменной, по каким-либо причинам, прочитать не получается, и в настройках модуля PL307 стоит признак, что на ответ при сбое возвращать ошибку-исключение 0xВ, то модуль вместо значения переменной добавляет в ячейку зарезервированные кодовые слова, которые позволяют выяснить причину недостоверности данных.

Кодовые слова могут быть следующего вида:

- NDEF – переменная еще ни разу не была опрошена модулем;
- NOANSW – внешнее устройство не отвечает;
- ERROR – внешнее устройство отвечает ошибкой на запрос чтения данной переменной.

Полный список всех архивных файлов с путями ведётся в файле list_file.xml. Этот файл расположен на SD-карте в корне папки /archive. Он доступен для скачивания по адресу: http://IP модуля/archive/list_file.xml. Формат данного файла представлен ниже:

```
<doc>
<file><year>2015</year><month>07</month><name>2015-07-01.csv</name></file>
```

```
<file><year>2015</year><month>07</month><name>2015-07-02.csv</name></file>  
...  
</doc>
```

Формат файла понятен из названия тэгов: значение тэга `<year>` - папка года, тэга `<month>` - папка месяца, `<name>` - имя файла, который вложен в эти папки. Имя файла включает год, месяц и день создания файла. Если в процессе работы архивации было: извлечение SD-карты, изменение таблицы переменных или изменение периода архивации, то всегда генерируется новый файл. Если файл на текущую дату был уже создан, то в имени файла включается нумерация для текущей даты. Например, у нас уже есть файл `2015-07-02.csv`, и в 13:00:00 часов вынули SD-карту и вставили ее обратно. Тогда модуль создаст новый файл с именем `2015-07-02_001.scv` и архивирование будет вестись уже в него, а старый файл закрывается. И так далее. Если количество файлов на одну дату превысит 256 (последний создаваемый файл `2015-07-02_255.scv`), то архивирование на текущую дату прекращается. И заново запустится только в 00:00:00 следующего дня.

Внимание! При эксплуатации модуля рекомендуется как можно меньше выполнять действий, приводящих к увеличению файлов на одну дату. Увеличение количества файлов увеличит нагрузку на производительность модуля при работе с архивами, что скажется на работе всей системы в целом!

Внимание! Если в настройках конфигурации модуля PL307 установлен пароль на основной сайт, то скачивание архивных файлов и файла `list_file.xml` возможно только после ввода пароля на главной странице основного сайта!

2.7.2 Настройка E-mail для уведомлений

Набор параметров для настройки E-mail уведомлений приведен на рисунке ниже. События, по которым отправляются уведомления, настраиваются на вкладке «Настройка событий» (см. п.2.8).

Всего можно настроить два адресата, которые будут получать уведомления по E-mail.

Отправка почты выполняется через внешний SMTP сервер. Модуль PL307 на данный момент не поддерживает шифрование, поэтому Тип передачи пароля и Тип соединения должны быть без шифрования. SMTP сервер должен поддерживать работу без шифрования.

Настройки оповещения по E-mail

Включить оповещение ?

IP/DNS – адрес SMTP-сервера ?

Адрес почтового ящика ?

Имя почтового ящика ?

Тема всех писем от модуля ?

Пароль для почтового ящика

Подтвердить пароль

Тип передачи пароля
Пароль без шифрования

Тип соединения
Без шифрования (порт 25)

Адрес первого почтового ящика (E-mail1) ?

Адрес второго почтового ящика (E-mail2) ?

Рисунок 2.15 – Настройки E-mail

2.7.3 Настройка синхронизации времени

Синхронизацию времени можно выполнить 2-мя способами:

- вручную задать дату и время с ПК. Для этого необходимо нажать кнопку «Задать время с ПК» на вкладке «Архивирование» на странице конфигурации. Скрипт, автоматически, текущее время из персонального компьютера запишет в модуль PL307. Если батарейка на модуле установлена, и она не разряжена, то заданные дата и время не сойдутся и продолжат правильно идти и после отключения питания модуля. Недостатки способа: дата и время на ПК перед записью в модуль должны быть установлены достаточно точно, нет возможности автоматической синхронизации, обязательное наличие батареи для часов реального времени.
- второй способ более надежный и, фактически, не требует обязательного наличия батареи для питания часов реального времени. Это установка и синхронизация времени модуля PL307 с NTP - сервером. IP-адрес или DNS-имя NTP-сервера настраиваются на вкладке «Архивирование». В таком режиме, после любого сброса модуль соединяется с NTP-сервером и по протоколу SNTP устанавливает себе точное время. Общее время от старта модуля до установки времени занимает в пределах 3-х – 5-ти секунд. Затем, каждые 10 минут, модуль, автоматически, синхронизирует свое время с NTP-сервером. Этот

способ более надежный, но необходимо подключение к Internet или настроенный в локальной сети NTP-сервер. Если NTP-сервер выбран из интернета (например: ru.pool.ntp.org), то необходимо убедиться, что 123 порт UDP не блокируется в роутерах, проху-серверах и шлюзах.

2.8 Журнал событий

2.8.1 Настройка событий

В модуле PL307 реализовано ведение журнала событий. Журнал сохраняется на SD-карте в формате csv. Каждое событие может принимать два состояния: сигнальное (активно) и несигнальное (не активно). В журнале регистрируется переход из несигнального в сигнальное состояние и наоборот.

Настройка списка событий выполняется на отдельной вкладке «Настройка событий» в виде таблицы. Все настройки событий сохраняются в файле «event_file.csv», который также можно редактировать вручную на компьютере. Файл «event_file.csv» хранится в папке sys\serv.

С каждым событием связывается уникальное имя тега, которое объявляется в этой же таблице. Значение тегов таблицы «События» может запрашиваться и использоваться наряду с тегами «Таблицы опроса».

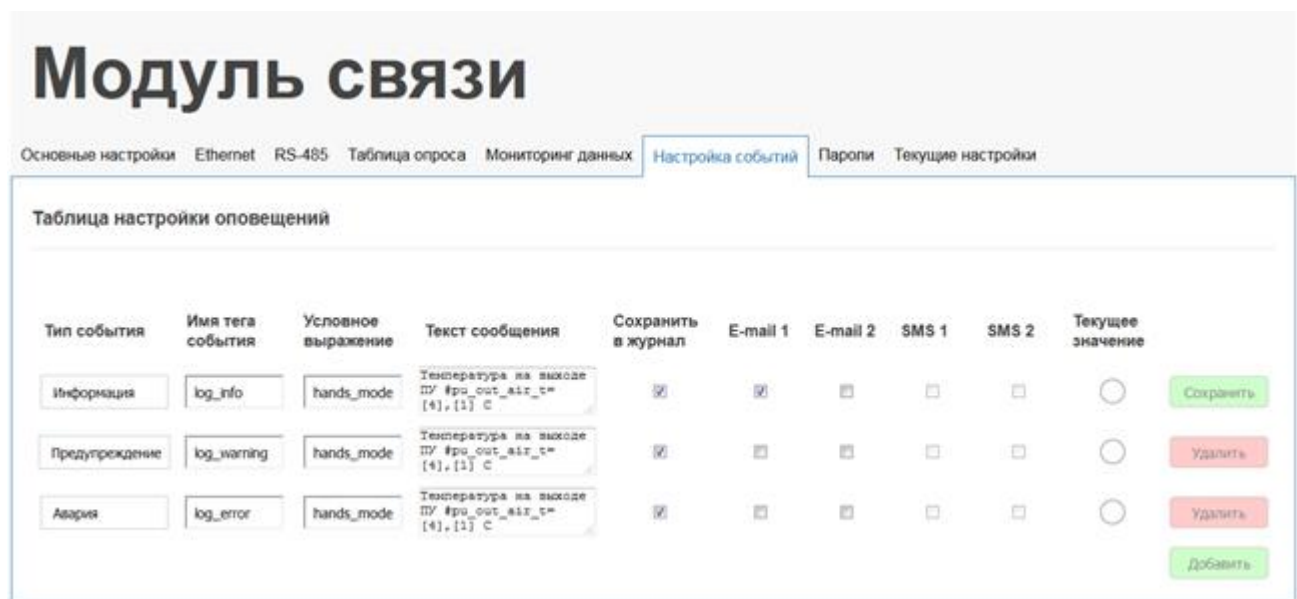


Рисунок 2.16 – Настройки событий

Пример файла «event_file.csv» приведен ниже.

```
0;log_info;hands_mode_autobits & 1;Температура на выходе ПУ #pu_out_air_t=[4],[1] C;1;0;0;0;0
1;log_warning;hands_mode_autobits & 2;Температура на выходе ПУ #pu_out_air_t=[4],[1] C;1;0;0;0;0
2;log_error;hands_mode_autobits & 4;Температура на выходе ПУ #pu_out_air_t=[4],[1] C;1;0;0;0;0
```

Вид таблицы в Microsoft Excel.

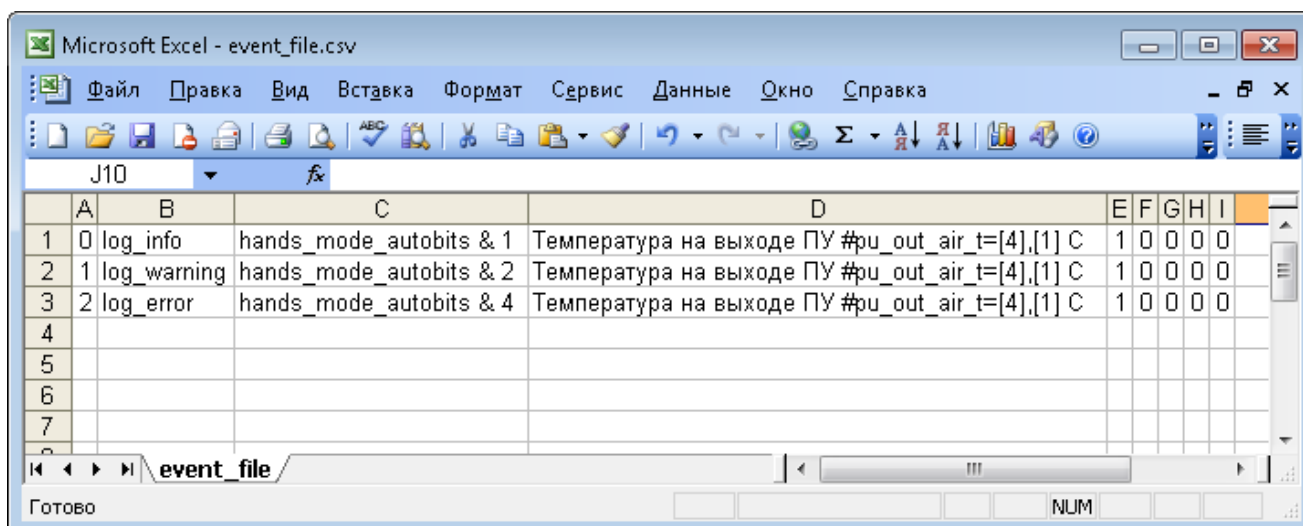


Рисунок 2.17 – Файл event_file.csv

Таблица 2.2 - Назначение полей при настройке событий

Столбец	Содержимое	Описание
A	Тип события	0- Информация, 1- Предупреждение, 2- Авария. Тип события может использоваться в программе верхнего уровня при отображении журнала событий
B	Имя тега события	Имя тега, с которым связывается текущее состояние события
C	Условное выражение	Условное выражение в синтаксисе Си. Событие считается сигнальным, если результат вычисления выражения больше 0. Значение выражения присваивается тегу из предыдущего столбца
D	Текст сообщения	Текст, который записывается в журнал при изменении состояния события. В текст можно вставлять форматированные значения любых тегов.
E	Сохранить в журнал	0- Не сохранять, 1- Сохранять. Если в этом поле поставить 0, то выражение будет вычисляться, присваиваться тегу, но в журнал сохраняться не будет. На отправку E-mail и SMS не влияет
F	E-mail 1	0- Не отправлять, 1- Отправлять. Если установлено в 1, то текст сообщения будет отправляться получателю E-mail 1
G	E-mail 2	Тоже самое для получателя E-mail 2
H	SMS 1	Тоже самое для получателя SMS 1
I	SMS 2	Тоже самое для получателя SMS 2

2.8.2 Форматирование строк при выводе событий в лог

При выводе строки в лог можно добавить в нее специальные символы для подстановки значений переменных в эту строку. Спецификатор команды подстановки начинается с символа # (решётка) после чего указывается имя переменной, значение которой необходимо подставить в строку лога. После имени переменной обязательно должен быть символ = (равно). Значения переменной можно форматировать для этого после знака равно необходимо добавить две квадратные скобки ([],[]). В первом поле между квадратными скобками указывается общая ширина поля в символах. Во втором - точность чисел с плавающей точкой. Общий вид команды подмены представлен в таблице ниже.

Таблица 2.3

#ident=[flag][width],[precision]	
#	символ места подстановки
ident	имя переменной, её значение будет подставлено в строку
=	символ окончания идентификатора
flag	символ + или(и) 0
width	ширина поля в символах
precision	точность

В случае если значение переменной ошибочно будет подставлена строка вида **ident=[строка ошибки]**. Спецификатор флаг символ + (плюс) позволяет добавить к положительным значениям переменных знак плюс. Спецификатор флаг символ 0 (ноль) позволяет заполнить все не используемые значением символы перед числом нулями.

Возможные ошибки переменной:

- TAG NEVER READ – модуль, в котором хранится значение переменной еще не опрашивался;
- TAG MODULE NO ANSWER – модуль, в котором хранится значение переменной не отвечает на запросы;
- TAG ERROR - модуль выдает ошибки;
- NOT SUPPORT ARRAY - подстановка переменной типа массив не поддерживается;
- NOT SUPPORT STR - подстановка переменной типа строка не поддерживается;
- TAG NOT FOUND - переменная не существует или PL307 работает в режиме шлюза.

Пример 1: `float pressure = 20.0`

```
[IN]Авария давление #pressure= Па
[OUT]Авария давление 20.000000 Па
```

Пример 2: `float pressure = 200.5879`

```
[IN]Авария давление #pressure=[+010],[2] Па
[OUT]Авария давление +000200.59 Па
[IN]Авария давление #pressure=[],[2] Па
[OUT]Авария давление 200.59 Па
```

Пример 3: `pressure =` переменной не существует

[IN]Авария давление #pressure= Па
[OUT]Авария давление pressure=[TAG NOT FOUND] Па

2.8.3 Файл журнала CSV

Файлы журналов пересоздаются каждые сутки (в 00:00), либо, если будет превышен предельный размер одного файла. Предельный размер файла задан жёстко в программе модуля и равен 1 МБ.

Файлы журнала хранятся в папках со строгой иерархией. Корневая папка журнала называется log и расположена в корне SD-карты. В папке log лежат папки с названием годов, т.е. 2017, 2018 и т.д.. В папках годов лежат папки месяцев 01, 02 ... 12. В папках месяцев лежат файлы журнала за текущий месяц. Наименование файлов журнала стандартизировано и имеет вид 2017-06-29_005.csv. Т.е. YYYY_MM_DD_NNN.csv. Здесь _NNN это порядковый номер файла журнала в пределах одних суток. Для самого первого файла _NNN опускается, т.е. будет выглядеть так 2017-06-29.csv, а следующим после него будет 2017-06-29_001.csv.

Формат файла:

```
Time;uTime;EType;EValue;Message
13:16:20.210;1498742180.21;0;1;Температура обратной воды ниже +10 °С (9.8 °С);
13:22:04.092;1498742524.092;0;0;Температура обратной воды ниже +10 °С (10.5 °С);
23:47:22.002;1498780042.002;2;1;Авария вентилятора;
```

Кодировка UTF-8. Здесь в первой строке наименования колонок, вторая и последующая строки записи журнала. Назначение колонок:

- Time - время события в пределах текущих суток в читаемом формате HH:MM:SS.MS, с точностью до мс;
- uTime - время события в пределах текущих суток в формате UTC unix time;
- EType - тип события, число (0- Информация, 1- Предупреждение, 2- Авария);
- EValue - Активно или Неактивно, число (0 - неактивно, 1 - активно);
- Message - текст сообщения, строка в формате UTF-8.

2.9 Пароли

Можно установить четыре пароля, которые используются в работе web-сервера:

- пароль для основного сайта – используется для ограничения доступа к основному содержимому пользовательского web-сайта;
- пароль для страниц с сервисными настройками – используется для защиты пользовательских страниц с сервисными настройками (защищенные страницы должны размещаться в папке \web\service\);
- пароль для страниц с заводскими настройками - используется для защиты пользовательских страниц с заводскими настройками (защищенные страницы должны размещаться в папке \web\vendor\);
- пароль для страницы конфигурации.

Нет имен пользователей, но анализируется сам пароль. При этом пароль от страниц конфигурации имеет доступ ко всем остальным пользовательским страницам, включая заводские и сервисные настройки. Пароль для страниц с заводскими настройками имеет доступ к сервисным настройкам и основному содержимому

пользовательского сайта. И пароль от страниц с сервисными настройками имеет доступ к основному содержимому пользовательского сайта.

Пароль «по умолчанию» для страницы конфигурации при установленном переключателе SW1 и для нового модуля – 111111.

Через 5 мин бездействия браузера (отсутствие запросов к модулю) признак введенного пароля модулем сбрасывается.

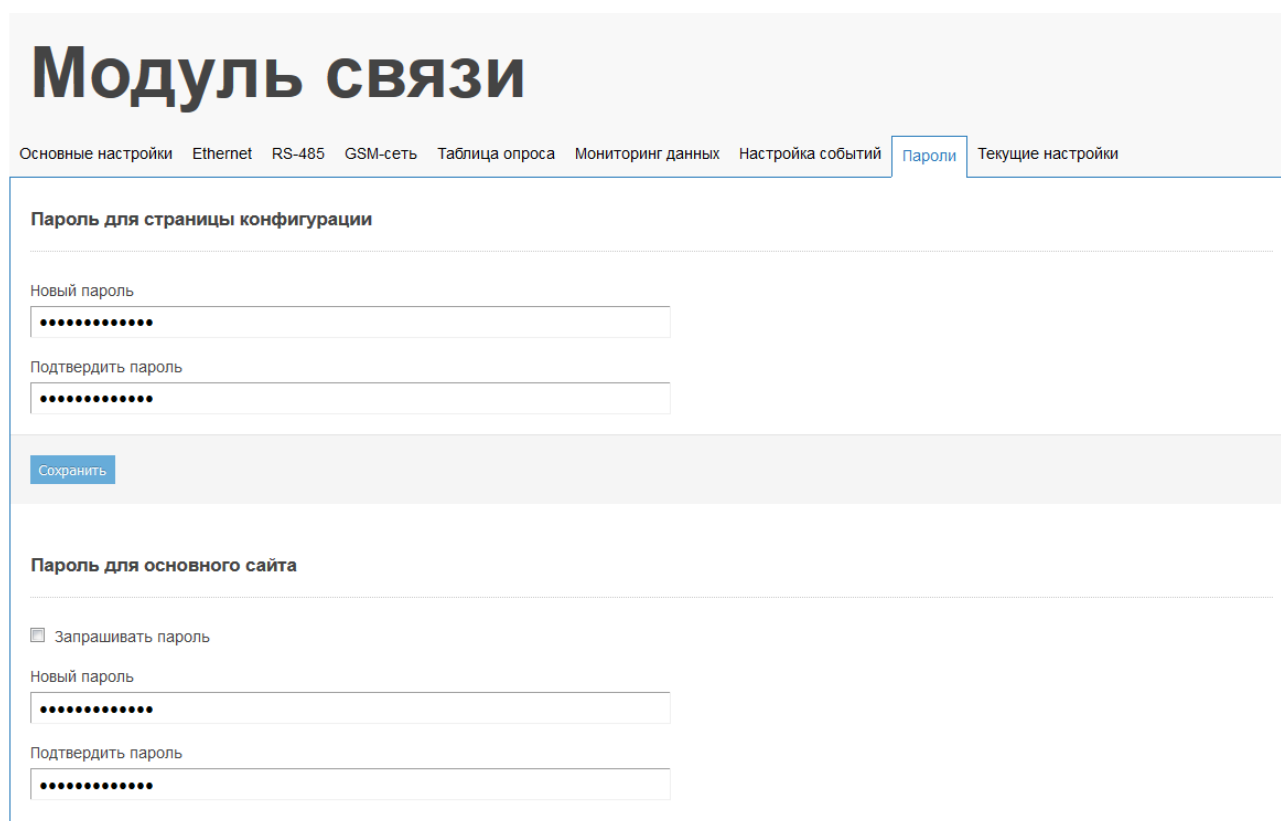


Рисунок 2.18 – Страница настроек паролей

2.10 Выполнение скриптов

2.10.1 Общая информация

Модуль PL307 поддерживает исполняемые скрипты, написанные с использованием синтаксиса языка Си. Скрипты описываются в виде функций с зарезервированными именами в файле script.c. Файл должен быть размещён на SD-карте в папке prog.

Скрипты предназначены для вычисления произвольных выражений, где в качестве переменных используются имена тэгов. Тэги описываются в Таблице опроса на странице конфигурации (либо альтернативно в файле fileio.csv). В таблице опроса в том числе можно описывать переменные типа local, которые можно использовать только в скриптах в качестве внутренних переменных. Если записывается значение в переменную-тэг, привязанную к регистрам типа holding или coil, то запись будет выполняться по протоколу Modbus и в регистры внешних устройств.

Все ошибки в синтаксисе или логике скрипта, найденные устройством, можно увидеть в логах со страницы конфигурации во вкладке «Текущие настройки» с указанием строки и описанием типа ошибки.

2.10.2 Ограничения по использованию языка Си для скриптов

Для скриптов есть ограничения по использованию синтаксических конструкций языка Си:

- Не поддерживаются указатели, структуры и массивы;
- Запрещено использовать циклы;
- Запрещено использовать вложенность операторов if() else;
- Запрещена конструкция switch() case: ;
- Запрещено использовать другие функции, кроме предустановленных;
- Запрещено использовать операнды ++ и --. Например, a++ ;
- Запрещено использовать конструкции типа x? y : z;
- Запрещено использовать #define, #enum и другие предпроцессорные директивы;
- Запрещено использовать библиотечные функции sin(), ln(), sprintf() и т.п.;
- Длина одной строки кода на Си не должна превышать 1024 байта.

2.10.3 Поддерживаемые возможности языка Си

- Поддерживаются все операнды, кроме ++ и --, в том числе и !, ~, ^, %, », *=, && и т.п.;
- Поддерживаются следующие форматы констант: 0x12345; 1.2345; 12345;
- Разрешается использовать операторы if() и else без вложенности. Длина строки с условием в скобках if не должна превышать 1024 символа. Повторять эти операторы в скрипте можно сколько угодно раз;
- Разрешается оператор return, который заканчивает текущее выполнение скрипта;

2.10.4 Общий формат файла script.c

Заготовка со всеми поддерживаемыми скриптами файла script.c выглядит следующим образом:

```
init()           // вызывается один раз при старте устройства
{
}
main()           //вызывается каждый рабочий цикл
{
}
timer_proc(1000) //периодический вызов раз в X мсек, где X (в данном примере
1000) может быть любым числом(но не меньше цикла модуля) в параметре функции
{
}
event_proc()     //вызывается каждый раз перед выполнением полного цикла
формирования событий
{
```

```
}  
port1_proc() //вызывается после опроса всех внешних устройств, подключенных к  
Port1/RS-485  
{  
  
}  
socket1_proc() //вызывается после опроса всех внешних устройств, подключенных  
к Socket1/TCP  
{  
  
}  
socket2_proc() //вызывается после опроса всех внешних устройств, подключенных  
к Socket2/TCP  
{  
  
}  
synch_proc() ///вызывается после опроса всех внешних устройств в режиме  
синхронного опроса  
{  
  
}
```

Модуль допускает отсутствие любой из этих функций, но с обязательной записью в лог, что данный скрипт отсутствует и не будет выполняться. Если не хочется загружать лог лишними сообщениями, то можно добавить в файл script.c ненужные скрипты, но оставить их пустыми.

2.10.5 Скрипт init()

Данный скрипт вызывается только один раз: при старте устройства, после вставки в разъем SD-карты; после сброса устройства или после модификации таблицы fileio.csv или devices.csv. Скрипт предназначен для выполнения инициализации переменных, используемых в других скриптах.

2.10.6 Скрипт main()

Данный скрипт выполняется каждый цикл модуля. Т.к. выполнение скрипта каждый цикл снижает производительность устройства - не рекомендуется данный скрипт перегружать вычислениями. Скрипт используется для отслеживания аварий или выполнения постоянных вычислений.

2.10.7 Скрипт timer_proc()

Скрипт таймера. В скобках задается период (в десятичной форме) в миллисекундах, с которым будет вызываться модулем данная функция. Если параметр задан 0 или не задан, то будет вызов раз в 1000 мсек (1 сек). Данный скрипт предназначен для выполнения различных временных задержек и других вещей, которые требуют отсчитывать определенные временные интервалы. Или выполнения проверок, которые не требуют быстрого постоянного контроля.

Пример, простейшего таймера на базе текущего скрипта:

```
timer_proc(1000)  
{  
  
    timer+=1;  
    //секундный таймер  
  
    if(!(timer%60))
```

```
    {  
        //минутный таймер  
    }  
    if(!(timer%3600))  
    {  
        timer=0;  
        //часовой таймер  
    }  
}
```

где, timer - локальная переменная в таблице переменных в файле fileio.csv

2.10.8 Скрипт event_proc()

Вызывается каждый раз перед новым циклом выполнения расчета настроенных событий в файле event_file.csv. Удобен для формирования масок ошибок, статусов. Отслеживания изменения переменных и т.п.

2.10.9 Скрипт port1_proc()

Вызывается каждый раз после полного цикла опроса внешних устройств, которые подключены к port1/rs-485. Опрос внешних устройств ведется согласно тегам, привязанным к данным устройствам. Скрипт удобен для выполнения первичной обработки данных с устройств, привязанных к port1 по RS-485.

2.10.10 Скрипт socket1_proc()

Вызывается каждый раз после полного цикла опроса внешних устройств, которые подключены к Socket1/TCP. Опрос внешних устройств ведется согласно тегам, привязанным к данным устройствам. Скрипт удобен для выполнения первичной обработки данных с устройств, привязанных к Socket1 по TCP.

2.10.11 Скрипт socket2_proc()

Вызывается каждый раз после полного цикла опроса внешних устройств, которые подключены на Socket2/TCP. Опрос внешних устройств ведется согласно тегам, привязанным к данным устройствам. Скрипт удобен для выполнения первичной обработки данных с устройств, привязанных к Socket2 по TCP.

2.10.12 Скрипт synch_proc()

Если в устройстве настроена синхронизация циклов опроса по всем каналам связи (когда каждый поток перед новым сканированием ждет завершения всех остальных), то данный скрипт будет вызываться, когда все потоки завершили опрос и готовы к новому сканированию. Скрипт удобен для выполнения общей обработки данных со всех внешних устройств. Если синхронизация потоков настроена не будет, то данный скрипт никогда не исполнится.

2.11 Логи модуля

Для отслеживания различных проблем и нестандартных ситуаций модуль PL307 ведет логи. Файлы с логами расположены в папке sys на SD - карте. SysLogPL.txt - основной лог, в который добавляются записи, а SysLogPL_old.txt - предыдущий лог. Посмотреть их можно как по ссылкам на странице конфигурации на вкладке "Текущие настройки",

так и путём прямой загрузки на компьютер с карты памяти microSD. В модуле реализована защита от неуместного разрастания файла с логами. После того, как основной файл с логами превысит размер 4 Мбайта, текущий лог полностью переписывается в файл с предыдущим логом с удалением в нем ранее записанной информации, а файл текущего лога очищается. Таким образом, файлы логов могут занять на SD-карте максимально 8 Мбайт.

2.12 Обновление прошивки модуля

Модуль позволяет обновить свою прошивку с помощью SD-карты или программы JL Configurator II.

Для обновления с помощью SD-карты необходимо выполнить ряд следующих действий:

- подключить к компьютеру карту microSD и убедиться, что она отформатирована как FAT32;
- в корне карты создать папку «system» и скопировать в нее актуальный файл прошивки PL307.bfm. Название папки и файла должно быть строго таким, иначе бутлоадер не найдет файл прошивки;
- установить переключатель SW2 в положение ON;
- вставить microSD-карту в разъем на устройстве;
- подать питание на устройство;
- если прошивка определилась, то светодиод перестанет мигать и остановится на одном из цветов или будет выключенным; индикация останавливается приблизительно на время 5-10 секунд (время стирания памяти микроконтроллера);
- если запись прошла успешно, то по истечении времени 5-10 секунд, светодиод опять замигает быстро попеременно красным или зеленым цветом.
- отключить питание модуля;
- переключатель SW2 вернуть в положение OFF;
- включить питание модуля;
- если прошивка обновилась верно, то светодиод начнет мигать приблизительно 2 раза в секунду зеленым или красным цветом.

3 Создание динамических сайтов

3.1 Общие сведения

Одной из основных функций модуля PL307 является Web-сервер. На SD-карте можно хранить любой сайт. В режиме шлюза сайт можно сделать только статическим и отображать какую-то общую неменяющуюся информацию, например сайт-визитку. В режиме сервера из сайта можно сделать HMI для контроля и управления вашей системой. Любую переменную, описанную в таблице опроса на странице конфигурации, можно отобразить на web-странице и настроить её динамическое обновление, используя технологию AJAX. Фактически, страницу в браузере можно сделать аналогичной странице в SCADA-системах, а модуль встроить непосредственно в ваше оборудование. Для работы требуется только обычный браузер, запущенный на любом устройстве – компьютер, планшет, смартфон. Из любых современных гаджетов можно получить доступ к вашему оборудованию, как из локальной сети, так и из глобальной Internet (при соответствующей настройке роутера).

Для работы понадобятся следующие программы: любой файловый менеджер (проводник. Total commander или аналогичные) и web-браузер кроме, IE и Яндекс.браузер (в них нельзя работать с таблицей опроса при ее формировании, если работы с таблицей через браузер не выполняется, то можно использовать любые браузеры). Также понадобятся: картридер, подключенный к ПК; microSD-карта, отформатированная под файловую систему FAT32.

Существуют следующие ограничения на работу с модулем PL307:

- microSD-карта должна постоянно находиться в модуле;
- начальная страница сайта должна называться index.html и никак иначе;
- нет поддержки стандартной работы форм (параметры формы должны обрабатываться и отсылаться с помощью javascript);
- количество задействованных MODBUS – регистров, описанных в таблице, не должно превышать:
 - для holding - не более 1024 регистров,
 - для Input - не более 256 регистров.
- ограничение по размеру запроса/ответа на групповое чтение переменных с помощью Ajax; чтобы не отслеживать сетевые TCP-пакеты и их размер, достаточно ограничить групповое чтение 10-ю переменными за раз;
- в групповой записи переменных через Ajax за один запрос можно записать не больше 4-х переменных.

Ниже описана технология создания встраиваемого сайта вручную. Также разработан специальный фреймворк, который позволяет значительно упростить создание сайта. Для получения фреймворка обратитесь в компанию ООО «Модуль Автоматика».

3.2 Создание сайта

При создании сайта необходимо обращать внимание на следующие ограничения:

- максимальный путь к файлу относительно корневой папки сайта, плюс имя файла с расширением, не должны превышать 300 символов;
- нет поддержки метода POST, PHP, MySQL, Perl и т.д.;

- поддерживаются только следующие типы файлов: «.html», «.htm», «.js», «.css», «.png», «.jpg», «.jpeg», «.bmp», «.gif», «.ico», «.swf». Файлы с другими расширениями будут доступны только для скачивания.

Желательно создать отдельную папку на локальном диске и в неё помещать все файлы, необходимые для работы встроенного сайта. Сайт всегда должен начинаться только со страницы «index.html». Допускается делать вложенные папки, главное, чтобы относительный путь и длина имени файла в сумме не превысили 300 символов. Если сайт будет статический, то достаточно создать нужное количество страниц, картинок и загрузить его на SD-карту. В случае необходимости сделать страницы, на которых данные должны изменяться в режиме реального времени, нужно задействовать специальные скрипты, написанные на языке JavaScript, позволяющие реализовать работу через технологию Ajax.

В браузерах есть специальный метод XMLHttpRequest. Он позволяет обновлять данные на странице Web-браузера адресно, без перерисовки страницы. Библиотека разработанных скриптов для работы с модулем PL307 использует этот метод. Механизм работы, следующий: браузер, формирует Ajax-запрос модулю PL307, модуль PL307 анализирует этот запрос и отвечает на него в формате JSON. Браузер разбирает полученные данные и отрисовывает их, согласно программе JavaScript.

Чтобы применить JavaScript-библиотеку, необходимо её подключить к странице, на которой будет происходить динамическое обновление. Файл с библиотекой называется ajax.js (текст файла приведен ниже). Подключение заключается в добавлении в заголовок html-страницы такой записи:

```
<script type="text/javascript" src="относительный путь/ajax.js"></script>
```

В библиотеке мы создаем глобальный объект xmlhttp типа XMLHttpRequest в функции Make_Ajax(). И работаем с этим объектом в дальнейшем коде.

В библиотеке три важных функции. Одна стандартная Read_Tag() и две с зарезервированным именем: Err_Ajax() и Sort_Data(). Стандартную функцию изменять не желательно, т.к. это может привести к потере работоспособности динамического обновления. Функции с зарезервированным именем можно менять по своему усмотрению, но в рамках определенных правил. Функция Err_Ajax() вызывается только в случаях ошибки работы Ajax: когда модуль не ответил или ответ от модуля является некорректным, браузер не поддерживает технологию Ajax или по другим ошибочным ситуациям. Эта функция нужна для отображения диагностики на странице сайта: например, написать текст об ошибке или поставить предупреждающую картинку и т.п. Если диагностика не нужна, то нужно оставить тело этой функции пустым. Полной противоположностью функции Err_Ajax() является функция Sort_Data(). Эта функция вызывается только после получения успешного пакета. В ней производим разбор принятых данных и отображаем их на странице. Также, в ней можно отображать статус успешного обмена с модулем PL307.

Рассмотрим правила создания Ajax-запроса для чтения или записи переменных с модуля PL307.

Запрос на чтение или запись создает и отправляет функция Read_Tag(). Вернемся к нашему примеру из темы о создании таблицы опроса. Запрос на чтение переменной, описанной в таблице, будет следующего вида:

```
/index.html?read=teg1
```

Если бы переменных было больше (например, еще были бы теги teg2, teg3, teg4), то их перечисляем в запросе через двойное подчеркивание. Запрос на чтение всех переменных выглядел бы следующим образом:


```
/index.html?read=teg1__teg2__teg3__teg4
```

И так можно читать за раз до 10 переменных. Читать переменные можно по одной или по несколько, главное не превышать размер запроса.

Запрос на запись выглядит следующим образом:

```
/index.html?write=teg1_1.234_teg2_11.00_teg3_2.4_teg4_123
```

Т.е. первым параметром идет символьный тег переменной, затем новое значение, потом следующая переменная и т.д. Ограничение на запись – 4 переменные за раз. Если нам нужно работать из скриптов javascript с массивами, то необходимо помнить, что каждый элемент массива разделяется символом «_» одиночное подчеркивание. Например, запись 7-ми элементов массива будет выглядеть следующим образом:

```
/index.html?write=arr1_1_2_3_4_5_6_7
```

Соответственно и при чтении, в ответе символьному имени arr1 будет присвоено значение 1_2_3_4_5_6_7.

3.3 Текст файла «ajax.js»

```
var xmlhttp=false;
//-----
//--- библиотеки -----
//-----
function Read_Tag(tag,timeout) {
    xmlhttp=false;
    xmlhttp=Make_Ajax();
    if(xmlhttp) {
        var url="/index.html?";
        url=url+tag;
        xmlhttp.open("GET", url, true);
        xmlhttp.timeout=timeout;
        xmlhttp.onreadystatechange = updatePage;
        xmlhttp.setRequestHeader("If-Modified-Since", "Sat, 1 Jan 2000 00:00:00
GMT");
        xmlhttp.send(null);
    }
    else
        Err_Ajax(); //браузер не поддерживает Ajax
}
function updatePage(){
    try {
        if(xmlhttp.readyState==4){
            if(xmlhttp.status==200)
                Sort_Data();
            else
                Err_Ajax();
        }
    }
    catch(e){Err_Ajax();}
}
function Test_Err_Module(val){
    if(val[0]=='?')
        return 0; // ошибка
    return 1; // все ОК
}
function Make_Ajax(){
    if (typeof XMLHttpRequest === 'undefined') {
        XMLHttpRequest = function() {
```

```
try { return new ActiveXObject("Msxml2.XMLHTTP.6.0"); }
catch(e) {}
try { return new ActiveXObject("Msxml2.XMLHTTP.3.0"); }
catch(e) {}
try { return new ActiveXObject("Msxml2.XMLHTTP"); }
catch(e) {}
try { return new ActiveXObject("Microsoft.XMLHTTP"); }
catch(e) {}
throw new Error("Не получается запросить данные с устройства!");
};
}
return new XMLHttpRequest();
}
```

3.4 Пример написания javascript для динамического обновления переменных на странице сайта

Для начала нам необходимо определиться, что мы хотим обновлять и с какой периодичностью. Например, если прочитать массив конфигурации и распределить его по соответствующим полям, то нам будет достаточно одного чтения переменных. Если мы читаем какие-либо статусы или важные переменные, то нам будет нужно скоростное циклическое обновление. И то, и другое делается стандартной функцией javascript. Называется она `setInterval()`. Если нам нужно делать циклическое обновление, то просто вызываем эту функцию. Если же, мы хотим прочитать данные единожды, то необходимо запомнить в глобальной переменной возвращаемый идентификатор, по которому мы, после удачного чтения, остановим выполнение таймера.

Вернемся к нашему примеру. Пусть у нас переменная `teg4` – это переменная конфигурации. А переменные `teg1`, `teg2`, `teg3` – данные с какого-либо датчика. Создаем файл, например, «`my.js`» в папке с сайтом. Добавляем этот файл в страницу для обновления наряду с файлом «`ajax.js`»:

```
<script type="text/javascript" src="относительный путь/my.js"></script>
```

Открываем файл. Для правильного разбора данных в функции `Sort_Data()`, нам необходимо будет создать еще 2 переменные: индекс начала обработки переменной и количество читаемых переменных. Эти переменные необходимы, чтобы попав в функцию-обработчик `Sort_Data()`, мы могли точно определить, что мы запрашивали и что сейчас будем обновлять. Объявляем переменные:

```
var ajaxidx,ajaxsc;
```

Раз мы хотим делать обновление переменной `teg4` единожды, а остальных в цикле, то нам понадобятся две функции, которые будут по таймеру вызываться и обновлять эти переменные. Назовем их `Read_Cfg()` и `Read_Data()`. Запускаем таймера для вызова этих функций.

```
window.setInterval(Read_Data,5);
var cfgtimerID=window.setInterval(Read_Cfg,100);
```

В `cfgtimerID` запомнили идентификатор таймера. Минимальное время обновления переменных в пределах 5 мсек. Но оно зависит от качества и скорости сети Ethernet. Создаем эти функции в нашем файле:

```
function Read_Data (){
    if(xmlHttp.readyState==0 || xmlHttp.readyState==4 || xmlHttp==false)
        {ajaxidx=0; ajaxsc=3; Read_Tag("read=teg1__teg2__teg3",1000);}
}
function Read_Cfg (){
    if(xmlHttp.readyState==0 || xmlHttp.readyState==4 || xmlHttp==false)
```

```
{ajaxidx=100; ajaxsc=1; Read_Tag("read=teg4",1000);}  
}
```

Как видно, функции простые. Проверяется, что сейчас не выполняется никаких Ajax-запросов и генерируется новый запрос, вызовом библиотечной функции `Read_Tag`. В функцию `Read_Tag()` передаются 2 параметра: текст Ajax-запроса и таймаут на ответ. Также, заполняем нужные индексы, которые нам понадобятся в функции `Sort_Data`. Для переменных `teg1`, `teg2`, `teg3` будут индексы, соответственно, 0, 1, 2. Для переменной `teg4` будет индекс 100. Индексы можно задавать любые.

Теперь создаем наши функции с зарезервированными именами `Sort_Data` и `Err_Ajax()`.

Так как ошибки отображать не будем, то тело `Err_Ajax()` оставляем пустым:

```
function Err_Ajax(){  
  
}
```

А в функции `Sort_Data` выполняем разбор принятых данных от модуля PL307

```
function Sort_Data(){  
    var r = JSON.parse(xmlHttp.responseText);  
    var old_ajaxidx=ajaxidx;  
    while(ajaxidx<(old_ajaxidx+ajaxsc)) {  
        switch(ajaxidx) {  
            case 0: document.getElementById('t1').innerHTML=r.teg1; break;  
            case 1: document.getElementById('t2').innerHTML=r.teg2; break;  
            case 2: document.getElementById('t3').innerHTML=r.teg3; break;  
            case 100:  
                window.clearInterval(cfgtimerID);  
                document.getElementById('CFG').innerHTML=r.teg4;  
                break;  
        }  
        ajaxidx++;  
    }  
}
```

Изначально, принятый текст в формате JSON необходимо разобрать. Это делает стандартная функция `JSON.parse()`. Мы получаем в переменной `r` структуру, в которой поля называются так же, как и символьные имена тегов, заданные в таблице опроса. Далее, выполняем цикл согласно переменным `ajaxidx`, `ajaxsc` и определяем по индексу с помощью конструкции `switch/case`, что мы читали. Как видно, что при первом удачном чтении переменной `teg4` (индекс 100), таймер запускающий функцию `Read_Cfg()` останавливается.

3.5 Работа с архивными данными

Для предоставления архивных данных, например для построения графиков в пользовательском сайте, в модуле PL307 реализован простой механизм, позволяющий просматривать и получать любые данные по всему архиву. Для этого необходимо задать: имя переменной, дату/время начала просмотра и длину просмотра. Формат запроса выглядит следующим образом:

```
http://IP/DNS адрес  
модуля/index.html?datatime=ГГГГММДДЧЧММСС&offset=4&value=имя
```

Либо так, если время необходимо указать в формате `unixtime`:

`http://IP/DNS адрес`
модуля/`index.html?udatetime=UUUUUUUU&offset=4&value=имя`

Где:

- `datetime` – в формате ГГГГММДДЧЧММСС. Календарная дата и время начала просмотра переменной с именем, заданным в `value`.
- `udatetime` – в формате UUUUUUUU. Дата и время начала просмотра переменной в формате `unixtime`.
- `offset` – длительность периода в секундах, за который нужно прочитать значения переменной с именем, заданным в `value`. Максимальный период: 1 сутки = $24 \cdot 60 \cdot 60$ секунд = 86400.
- `value` – имя переменной из таблицы опроса переменных на странице конфигурации.

Ответ от модуля приходит в JSON-формате следующего вида:

```
{"имя": [{"t": "UUUUUUUU", "v": "33.400002", "s": "0"}, ... , {"t": "UUUUUUUU", "v": "33.500000", "s": "0"}], "cnt": "6"}
```

Где:

- `Имя` – название массива, совпадающее с именем переменной из запроса.
- `cnt` – реальное количество элементов в массиве.
- `t` - временная метка в формате `unixtime` фактического сохранения.
- `v` – значение переменной. Если делитель `u` переменной не 1 или тип переменной `float`, то значение возвращается в формате `float` с учетом делителя. Если делитель 1, то для целых форматов (UINT16, SINT16, UINT32, SINT32) число возвращается, как есть, без форматирования, чтобы можно было обращаться к битам переменных.
- `s` – статус переменной: 0 – все ОК, 1 – переменная не была прочитана, 2 – модуль не отвечал, 3 – ошибка переменной. Если статус не 0, то значение переменной – недостоверно.

Алгоритм чтения архива для графиков выглядит следующим образом:

1. Определяется дата/время `dt` и диапазон `len`, исходя из которых, будем строить график для переменной `name`.

2. С помощью AJAX – запроса формируем запрос:

```
http://IP/DNS модуля/index.html?datetime=dt&offset=len&value=name
```

3. Разбираем полученный ответ функцией `JSON.parse`.

4. Проверяем, что поле `cnt` в ответе не 0. Если оно равно 0, то по данной переменной `name` в архиве за заданный промежуток ничего не было сохранено. Выходим из цикла чтения архива.

5. Запоминаем метку времени `t` у последней структуры массива `name[cnt-1].t`

6. Проверяем ее. Если `name[cnt-1].t=dt+len`, то архив за весь промежуток прочитан. Выходим из цикла чтения архива.

7. Если равенство не выполняется, то модифицируем переменные:

`len=len-(name[cnt-1].t-dt)` и `dt=name[cnt-1].t` и повторяем алгоритм, начиная с пункт 2.

8. Если нужно строить несколько графиков, то алгоритм с 1 по 7 пункт нужно повторить столько раз, сколько переменных, необходимо отобразить на графиках.

3.6 Загрузка сайта в модуль PL307

После того, как сайт полностью наполнен в папке на локальном диске, необходимо загрузить этот сайт в модуль PL307. Для этого понадобится файловый менеджер и картридер на ПК. Вставить SD-карту в картридер. В корне карты создать папку web. И в нее скопировать созданный сайт. Вставить карту в модуль. Сайт будет доступен по адресу: `http://IP-адрес`. Если сайт не будет открываться, то необходимо убедиться на странице конфигурации, что включен web-сервер и на какой порт он настроен. Если порт не 80, то в запросе браузера необходимо его явно указывать. Например, пусть порт будет 555. Тогда сайт будет открываться по следующей ссылке: `http://IP-адрес:555`, а страница конфигурации по адресу `http://IP-адрес:555/cfgPL/index.html`

На сайт можно установить защиту паролем. Пароль задается на странице конфигурации Вкладка «Пароль». В блоке пароль для основного сайта, устанавливаем галочку «Запрашивать пароль», задаем пароль на сайт и его подтверждение, затем нажимаем кнопку «Сохранить». Защита паролем сразу же будет активирована.

4 Техническое обслуживание

4.1 Общие указания

Техническое обслуживание устройства должно производиться обслуживающим персоналом не реже одного раза в шесть месяцев и включает в себя следующие операции:

- визуальный осмотр;
- очистку корпуса прибора и разъемов от пыли, грязи и посторонних предметов;
- проверку качества подключения кабелей.

Замену батарейки часов по п.4.4 необходимо выполнять один раз в 3 года, либо после длительного хранения или простоя без напряжения питания.

4.2 Меры безопасности

Любые работы по техническому обслуживанию (очистка и проверка качества подключений кабелей) и замене батарейки часов производить только при отключении устройства от источника питания.

4.3 Порядок технического обслуживания изделия

4.3.1 Визуальный осмотр

4.3.2 Очистка корпуса устройства и разъемов от пыли, грязи и посторонних предметов

4.3.3 Проверка качества подключения кабелей

4.4 Замена батарейки часов

Тип батарейки CR2032.

5 Хранение и транспортировка

5.1 Транспортирование

5.1.1 Транспортирование устройства в упаковке допускается при следующих условиях:

температура воздуха от - 20°C до +75°C;
относительная влажность воздуха не более 95% при температуре 35°C.

Транспортирование допускается всеми видами закрытого транспорта.

5.2 Хранение

5.2.1 Хранение устройства в упаковке допускается при следующих условиях:

температура окружающего воздухаот +5 до +40°C;
относительная влажность воздуха не более 80 % при температуре 25°C.

6 Гарантийные обязательства изготовителя

Изготовитель гарантирует соответствие устройства заявленным характеристикам при соблюдении условий эксплуатации, транспортирования, хранения и монтажа.

Гарантийный срок эксплуатации – 24 месяца со дня продажи.

Настоящая гарантия не действительна в случаях, когда повреждения или неисправность вызваны пожаром или другими природными явлениями; механическими повреждениями; неправильным использованием; ремонтом или наладкой, если они произведены лицом, которое не имеет сертификата, подтверждающего наличие знаний для оказания таких услуг, а также эксплуатацией с нарушением технических условий или требований безопасности.

В том случае, если в течение гарантийного срока часть или части устройства были заменены частью или частями, которые не были поставлены или санкционированы изготовителем, а также были неудовлетворительного качества и не подходили для товара, то потребитель теряет все и любые права настоящей гарантии, включая право на возмещение.

В случае выхода устройства из строя в течение гарантийного срока при соблюдении пользователем условий эксплуатации, транспортирования, хранения и монтажа предприятие-изготовитель обязуется осуществить его бесплатный ремонт или замену.

Устройство является технически сложным изделием, его ремонт осуществляется на предприятии изготовителя.

ВНИМАНИЕ! Для осуществления ремонта необходимо предоставить паспорт на изделие с отметкой о продаже. Без отметки о продаже с печатью или штампом продавца дата гарантии считается от даты изготовления.

ООО «Модуль Автоматика»
Россия, г. Пенза
+7 (8412) 98-10-14 (многоканальный)

info@maautomatics.ru www.mautomatics.ru