

## Руководство по эксплуатации блока GP4I-4O входов/выходов общего назначения стыка ModBus

### 1. ВВЕДЕНИЕ В MODBUS И ИЗДЕЛИЕ

- 1.1 Принцип действия
- 1.2 Основные характеристики

### 2. ИНДИКАТОР ПЕРЕДНЕЙ ПАНЕЛИ

- 2.1 Светодиоды Power, Rx и Tx
- 2.2 Переключатель PROG

### 3. СОЕДИНЕНИЯ

- 3.1 Электрические соединения
- 3.2 Соединения шины
  - 3.2.1. Соединения входов с сухими контактами
  - 3.2.2. Стандарт RS-485 и соединения цифровых входов

### 4. ХАРАКТЕРИСТИКИ КАБЕЛЕЙ

- 4.1 Электрические характеристики кабелей
- 4.2 Стандарт RS-485 и цифровые входы

### 5. ИЗОЛЯЦИЯ ШИНЫ И СОГЛАСУЮЩИЕ РЕЗИСТОРЫ

### 6. ОПИСАНИЕ MODBUS

- 6.1 Поддерживаемые функции
- 6.2 Исключительные реакции

### 7. ОПИСАНИЕ УСТРОЙСТВА И КОНФИГУРАЦИЯ

- 7.1 Общее описание
- 7.2 Схема конечного автомата
- 7.3 Настройка GPIO
  - 7.3.1 Настройка адреса
  - 7.3.2 Настройка связи
  - 7.3.3 Установка заводской конфигурации с помощью переключателя PROG

#### 7.4 Широковещание

#### 7.5 Сторожевой таймер

- 7.5.1 Время сторожевого таймера
- 7.5.2 Конфигурация сторожевого таймера
- 7.5.3 Состояние реле WDT

### 8. ЦИФРОВЫЕ ВХОДЫ

- 8.1 Конфигурация фильтра цифрового входа
- 8.2 Чтение влажных контактов

### 9. СЧЕТЧИКИ

- 9.1 Счетчики цифровых входов
- 9.2 Счетчик питания
- 9.3 Счетчик WTD

### 10 РЕЖИМЫ РАБОТЫ

#### 10.1 ОСНОВНОЙ РЕЖИМ

- 10.1.1 Заранее заданное начальное состояние реле
- 10.1.2 Проверка текущего состояния
- 10.1.3 Запрос изменения реле
- 10.1.4 Проверка регистра фиксируемых тревог
- 10.1.5 Проверка регистра мгновенных тревог

### 11. КАРТА ОСНОВНЫХ РЕГИСТРОВ MODBUS RTU

## 12. ВЕРСИЯ ИЗДЕЛИЯ

---

# 1. ВВЕДЕНИЕ В MODBUS И ИЗДЕЛИЕ

Благодарим вас за то, что вы приобрели GP4I-4O со свойствами MODBUS-RTU. Это руководство предназначено для профессионального установщика, если же вы им не являетесь, то, пожалуйста, проконсультируйтесь у вашего официального дистрибьютора. Далее, говоря о входах/выходах общего назначения (GPIO), мы будем подразумевать GP4I-4O.

MODBUS – это шина открытого поля, успешно используемая во всем мире для соединения полевых устройств с главным контроллером. Поэтому мы выбрали MODBUS, чтобы предложить нашим клиентам и партнерам автоматизированное решение, которое легко объединяется не только с нашими фирменными продуктами, но и с громадным ассортиментом компонентов и контроллеров третьих сторон.

MODBUS, MODBUS-RTU и остальные связанные названия являются зарегистрированными торговыми марками MODBUS Organization. Дополнительную информацию и документацию можно найти на сайте <http://www.Modbus.org/>

## 1.1 ПРИНЦИП РАБОТЫ

GPIO реализует MODBUS-RTU в качестве свойства управления-связи, которое позволяет выполнять его задачи эксплуатации и оперативного контроля из среды автоматизации MODBUS. Профилактическое техническое обслуживание и анализ отказов также возможны благодаря реализации внутренних регистров в GPIO с наиболее важными рабочими и ошибочными событиями.

Используя сообщение MODBUS-RTU, GPIO может управлять выходными реле, считывать цифровые дискретные входы, составлять отчеты об ошибках, исторических данных и т.д., предоставляя установщику/пользователю широкий ассортимент новых свойств на основе автоматизации.

## 1.2 ОСНОВНЫЕ ХАРАКТЕРИСТИКИ

Система связи MODBUS обеспечивает реализацию «Ведущий/Ведомый» среди устройств, совместно использующих физическое соединение. Физическим соединением для GPIO является полудуплексный последовательный слой RS485, который был выбран из других вариантов благодаря его широкому распространению и простоте.

Для GPIO было реализовано полудуплексное проводное соединение RS-485, а излучатель предназначен для работы в системе с одним ведущим устройством. В этой реализации Ведущий и Ведомый играют ясную роль, которую важно точно понимать для надлежащей реализации системы.

Ведущее устройство: Устройство, которое управляет обменом данными в шине и при необходимости реализует задачи координации между различными ведомыми (то есть, программируемым логическим контроллером (ПЛК), системой SCADA и т.д.).

Ведомое устройство: Устройства, подсоединенные к шине, которые обслуживают запросы ведущего и сообщают информацию или выполняют задачи по запросу Ведущего.

# 2. ИНДИКАТОР ПЕРЕДНЕЙ ПАНЕЛИ

Индикатор панели модуля GPIO имеет три светодиода и один переключатель. Эти элементы предназначены для воздействия на устройство и отображения его состояния. На рисунке 1 показан индикатор панели.

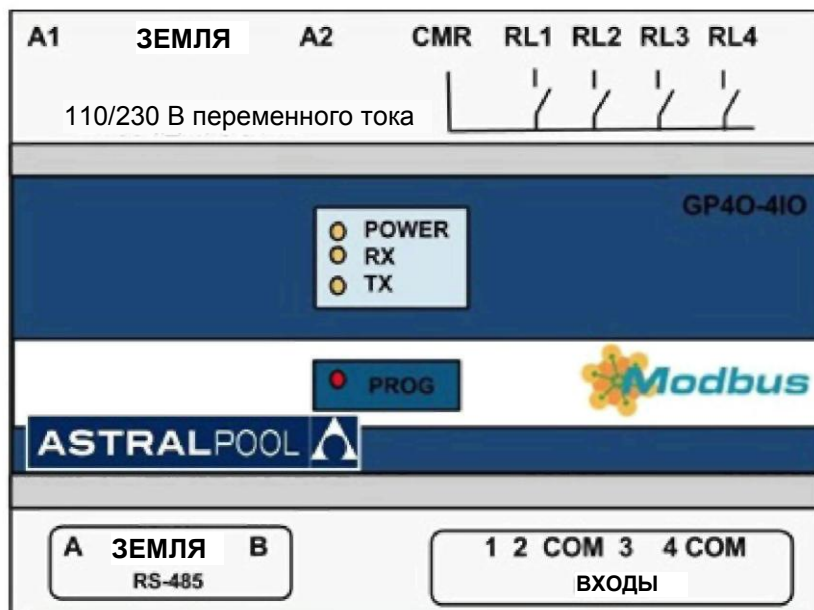


Рисунок 1. Индикатор панели

## 2.1 Светодиоды Power, Rx и Tx

Эти три элемента работают следующим образом:

- Светодиод POWER (ПИТАНИЕ) ВКЛЮЧЕН, если на модуль подано питание, в противном случае он будет ВЫКЛЮЧЕН.
- Светодиод RX (Прием) будет мигать, когда модуль GPIO получает данные.
- Светодиод TX (Передача) будет мигать, когда модуль GPIO посылает данные.

Rx и Tx являются соответственно сокращениями слов “receiving” («прием») и “transmitting” («передача»). Как только ведомый обнаруживает пакет от ведущего, светодиод Rx начинает мигать, а если ведомый реагирует на этот запрос, то будет мигать светодиод Tx.

## 2.2 Переключатель PROG

Кнопка Prog предназначена для ручного сброса устройства в заводские настройки и расположена под светодиодами; процедура сброса описана в разделе 7.3.3 «Установка заводской конфигурации с помощью переключателя PROG».

## 3. СОЕДИНЕНИЯ

### 3.1 Электрические соединения

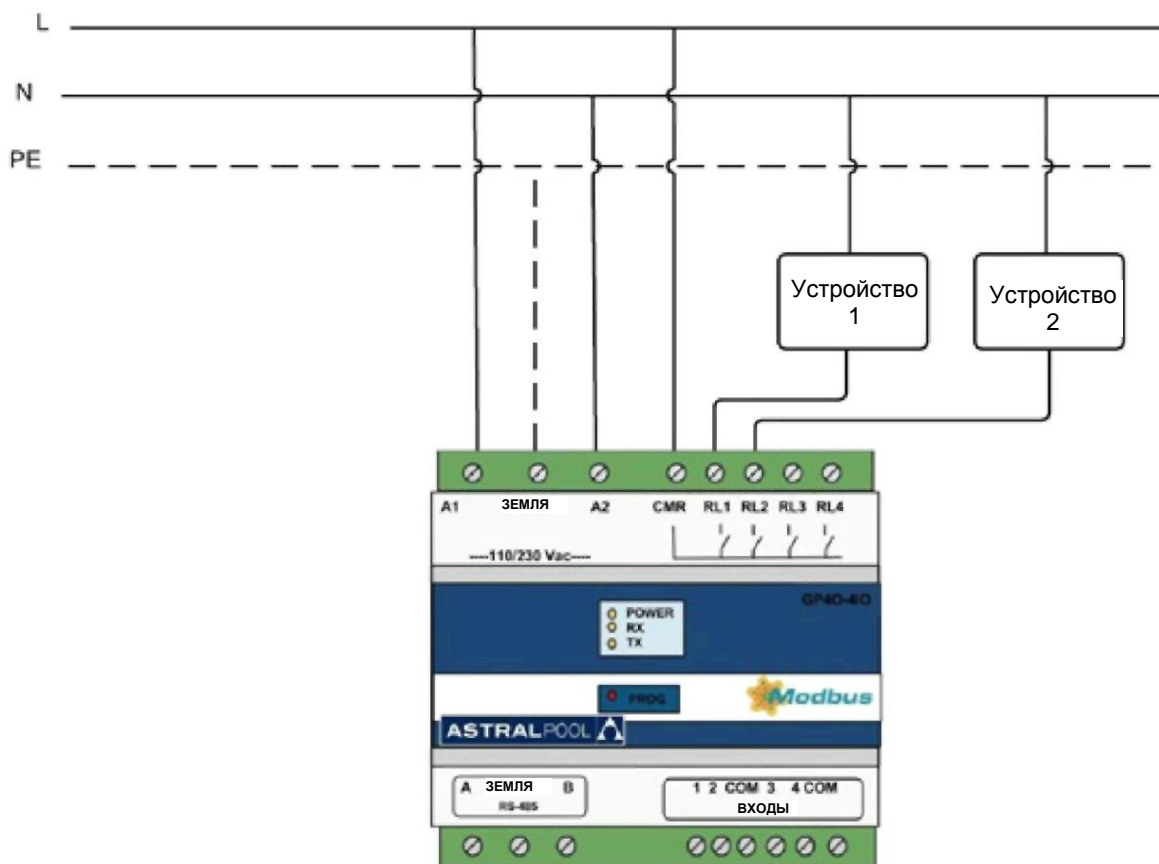


Рисунок 2. Электрическое соединение

Реле GPIO могут одновременно разрешать или запрещать ток четырех устройств. Однако требуется предпринять некоторые меры предосторожности, чтобы гарантировать целостность GPIO.

1. Все выходы должны питаться одинаковым напряжением (клемма CMR на рисунке 2), никогда не смешивать устройства с разным напряжением питания, эта ошибка может привести к разности потенциалов между различными выходами и к чрезмерному или недостаточному питанию их, вызывая неустраняемые отказы, несмотря на внутреннюю защиту устройств. Номинальное выходное напряжение  $V_n=250$  В.
2. Аналогично дополнение токов реле не должно превышать номинальный ток ( $I_n=5$  А). В противном случае реле будут повреждены.
3. Если GPIO контролирует различные устройства, то ток должен отрезаться на одинаковой стороне фазы **во всех устройствах**. Если это не так, то фаза и нейтраль встретятся **внутри GPIO**, вызывая серьезные повреждения реле.

Примечание: Возьмите эти три совета и внедрите их в свою рабочую методологию, когда потребуется устанавливать GPIO, на рисунке 2 показан пример двух контролируемых устройств.

### 3.2 Соединения шины

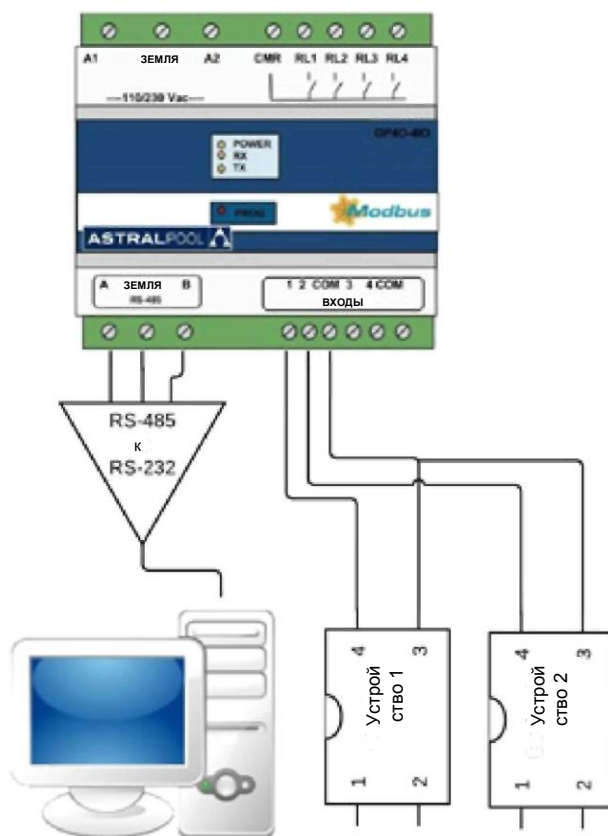


Рисунок 3. Соединение шины

### **3.2.1. Соединения входов с сухими контактами**

GPIO имеет четыре реализованных цифровых дискретных входа, которые не имеют связанного напряжения (сухие контакты) и могут обнаруживать цифровые импульсы. Эти импульсы могут считываться, а их задние фронты сохраняться, а также реализован цифровой фильтр, чтобы гарантировать правильное измерение. Если некоторые сигналы с напряжением (влажные контакты) должны считываться, то они могут использовать **оптоизолятор**. В разделе 8 «Цифровые входы» описаны все эти свойства.

### **3.2.2. Стандарт RS-485 и соединения цифровых входов**

С другой стороны, RS-485 обеспечивает стандартные связи ANSI/TIA/EIA-485. Некоторые производители меняют местами соединения «А» и «В», что может вызвать замешательство, но не в случае GPIO. Помните об этом при подключении GPIO к устройствам MODBUS других компаний.

## **4. Характеристики кабелей**

### **4.1 Электрические характеристики кабелей**

Так как GPIO питается от электросети, то не имеет значения, что линия (L, черный, коричневый или серый) подходит к клемме A1, а нейтраль (N, синий) – к клемме A2, или наоборот. Защитная земля/заземление (PE, желтый и зеленый) должна быть соединена с клеммой GND. Это устройство может питаться от любой электросети, проверьте руководство BRON\_GPIO\_62368\_v1.0\_EN для получения дополнительной технической информации.

### **4.2 Стандарт RS-485 и цифровые входы**

Рекомендуемая проводка для связи MODBUS-RTU основана на линейной структуре, активной шине с согласованием на обоих концах. Возможна стыковка и расстыковка устройств во время работы без влияния на другие устройства. Провод должен быть скручен и экранирован в соответствии с EN 50 170.

Значения скорости передачи, поддерживаемые

устройством, максимальная длина кабеля 1200 метров без ретрансляторов или до 10 километров с ретрансляторами – при установке в соответствии со стандартом.

Для сбалансированной пары, используемой в системе RS485, может быть предпочтительным характеристический импеданс со значением, превышающим 100 Ом, особенно при скорости передачи 19200 бод и выше.

## 5. ИЗОЛЯЦИЯ ШИНЫ И СОГЛАСУЮЩИЕ РЕЗИСТОРЫ

Если связанная шина доступна для пользователя, то она должна быть дважды изолирована. Так как обычно доступность шины для пользователей будет зависеть от каждой конкретной установки, то защитная изоляция была реализована на физическом уровне шины GPIO. Кроме того, в целях безопасности рекомендуется обеспечить, чтобы остальные устройства, совместно использующие эту шину, также реализовали эту изоляцию. Наконец, использование устройств с изолированной шиной не только улучшает уровень защиты, но и увеличивает надежность оборудования, большую устойчивость к электромагнитным помехам, более высокую надежность, большую стабильность на всем диапазоне температур. Вне зависимости от того, одно или несколько устройств подсоединены, совместно используя физическое соединение шины, рекомендуется использовать согласующие резисторы на концах шины, что особенно важно при использовании кабеля большой длины и при высоких скоростях обмена данными. Согласующий резистор используется для предотвращения отражения радиочастотного сигнала от конца шины, вызывающего помеху. Согласующие резисторы должны находиться на обоих концах шины, будучи соединенными параллельно (как показано ниже на рисунке). Типовое значение сопротивления этого резистора 120 Ом, мощность 0,5 Вт. Значение сопротивления резистора должно быть одинаковым на обоих концах. Согласующие резисторы – это резисторы  $R_T$  на рисунке 4 «Соединение устройств».

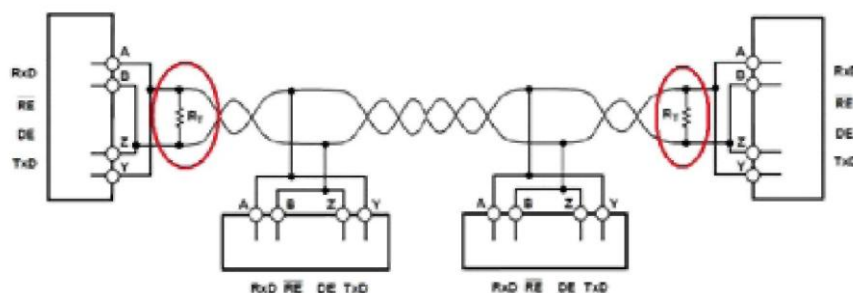


Рисунок 4. Соединение устройств

## 6. ФУНКЦИИ MODBUS

### 6.1 ПОДДЕРЖИВАЕМЫЕ ФУНКЦИИ

Пожалуйста, будьте осторожны при возможных включениях и убедитесь, что используемая функция является правильной.

#### РЕЖИМ БИТОВОГО ДОСТУПА

Функции в режиме битового доступа реализуются в соответствии со стандартом MODBUS-RTU, описанным в файле [http://www.Modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b.pdf](http://www.Modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf).

0x01 ЧТЕНИЕ КАТУШЕК.

0x0F ЗАПИСЬ НЕСКОЛЬКИХ КАТУШЕК.

0x05 ЗАПИСЬ ЕДИНИЧНОЙ КАТУШКИ.

0x02 ЧТЕНИЕ ДИСКРЕТНЫХ ВХОДНЫХ СИГНАЛОВ.

#### РЕЖИМ РЕГИСТРОВОГО ДОСТУПА

Функции в режиме регистрового доступа реализуются в соответствии со стандартом MODBUS-RTU, описанным в файле [http://www.modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b3.pdf](http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf).

Обычно регистры являются 16-разрядным кодом без знака.

0x03 ЧТЕНИЕ РЕГИСТРОВ ХРАНЕНИЯ

0x04 ЧТЕНИЕ ВХОДНЫХ РЕГИСТРОВ

0x10 ЗАПИСЬ НЕСКОЛЬКИХ РЕГИСТРОВ

0x06 ЗАПИСЬ РЕГИСТРА ХРАНЕНИЯ

### 6.2 ИСКЛЮЧИТЕЛЬНЫЕ РЕАКЦИИ

Исключительные реакции реализуются в соответствии со стандартом MODBUS-RTU, описанным в файле [http://www.Modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b3.pdf](http://www.Modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf).



Реализованы исключения от 1 до 4.

Исключение 4 появляется, если кадр пытается писать счетчик питания или счетчик WTD, описанные в разделе 11 «Карта основных регистров ModBus RTU». Остальные коды исключения (от 1 до 3) появляются в нормальных условиях.

## 7. ОПИСАНИЕ УСТРОЙСТВА И КОНФИГУРАЦИЯ

### 7.1 ОБЩЕЕ ОПИСАНИЕ

Через запрос в MODBUS можно управлять четырьмя реле GPIO, а реальное состояние его четырех цифровых входов может считываться.

Если одновременный запрос выполняется через Modbus и дистанционное управление, то конфликт не возникает, так как будет действовать последний запрос.

Обычно не проверяется логичность значений, посылаемых в конкретные регистры. Поэтому оператор несет ответственность за проверку этой логичности.

В этом руководстве числа в шестнадцатеричной форме представляются в формате **0xZZ**, где ZZ – это число. Входной регистр, который управляет схемой конечного автомата, показан в разделе 7.2, а все ячейки памяти описаны в разделе 11 «Карта основных регистров ModBus RTU».

### 7.2 СХЕМА КОНЕЧНОГО АВТОМАТА

При ВКЛЮЧЕНИИ питания системы состоянием всегда является «Старт». Если сторожевые таймеры не сработают, и никакой запрос не выдается после ВКЛЮЧЕНИЯ питания, то система будет оставаться в состоянии «Старт».

Из состояния «Старт» возможен переход в состояние «Запрос» (если выдан некоторый запрос) или состояние «Сторожевой таймер» (если он сработал). Из состояния «Запрос» мы можем снова перейти в состояние «Запрос», если делается другой запрос. Если выработано состояние «Сторожевой таймер», то оно может изменить состояние только посылкой запроса, или если настройка сторожевого таймера подходит для этого, то будет достигнуто состояние «Старт» (показано пунктирной стрелкой). На рисунке 5 показаны эти переходы.

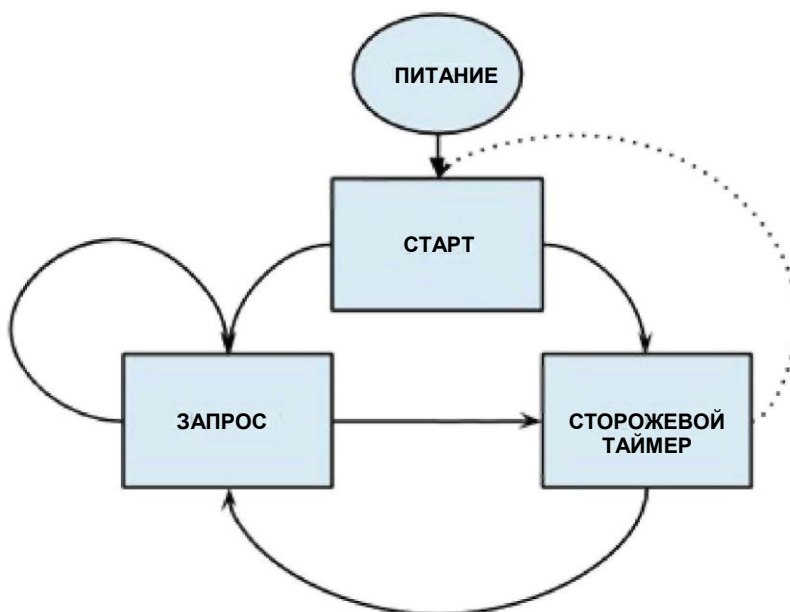


Рисунок 5. Схема состояний

### 7.3 Настройка GPIO

#### 7.3.1 Настройка адреса

Адрес GPIO в шине устанавливается через регистр хранения 0x00.

ID\_Address: Адрес GPIO в шине.

Заводская настройка: 0x1A.

Рекомендуемый диапазон значений: 0x1A - 0x1E.

Заводская настройка адреса GPIO, принятая по умолчанию, – это 0x1A. Однако если должно быть реализовано несколько GPIO, то значения ID\_Address не должны совпадать.

Этой ситуации можно избежать путем записи в

регистр хранения 0x00. В следующем примере показано, как выполнять эту процедуру.

Пример: изменить ID\_Address с 0x1A (по умолчанию) на 0x1B.

Передаваемое сообщение: 1A 10 00 00 00 01 02 00 1B 58 AB

Где:

- 1A это адрес ведомого (реальный ID\_Address).
- 10 это используемая функция. Запись нескольких регистров.
- 00 00 это адрес регистра хранения для записи.
- 00 01 это количество регистров, подлежащих записи (1 в данном случае).
- 02 это количество байтов данных, подлежащих передаче.
- 00 1B это новый ID\_Address.
- 58 AB это контрольная сумма избыточного циклического контроля (CRC).

### 7.3.2 Настройка связи

Выбор скорости обмена последовательной связи осуществляется через регистр хранения 0x01. По умолчанию, это 9600 бод и значение 8E1 (8 бит данных, четность, 1 стоповый бит). Однако поддерживаются также другие настройки, показанные в следующей таблице настроек связи, поддерживаемых со стороны GPIO.

Причина поддержки кадров N2 заключается в сохранении требования стандарта MODBUS о посылке одиннадцати битов на каждый байт (1 стартовый бит + 8 битов данных + 1 бит четности + 1 стоповый бит). Если выбрана конфигурация без проверки четности, то вводятся 2 стоповых бита для сохранения одиннадцати битов на каждый байт, требуемых стандартом.

Из соображений совместимости поддерживаются также кадры N1. Однако помните, что используя этот вариант, вы не выполняете требования стандарта MODBUS, так как используются только десять битов на каждый байт.

В соответствии с этим, выбор скорости обмена и кадра выполняется определением скорости обмена (в бодах), количества битов данных, четности и количества стоповых битов.

COM_Setup:	Настройка связи	
Заводская настройка:	0	9600, 8E1
Поддерживаемые значения:	0	9600, 8E1
	1	19200, 8E1
	2	9600, 8N2
	3	19200, 8N2
	4	9600, 8N1
	5	19200 8N1



### 7.3.3 Установка заводской конфигурации с помощью переключателя PROG

Следующий пример служит иллюстрацией процедуры сброса GPIO для восстановления оригинального ID\_Address. Знайте, что значения регистров DI\_1\_counter, DI\_2\_counter, DI\_3\_counter, DI\_4\_counter и регистров фиксируемых тревог будут стерты в течение этого процесса (эти регистры описаны в разделах 9.1 и 10.1.4, соответственно). Для выполнения сброса в заводскую конфигурацию выполните следующие действия.

1. Отключите устройство от электросети.
2. Нажмите кнопку PROG.
3. Удерживая нажатой кнопку PROG, снова подключите GPIO к электросети.

GPIO вернется в «начальное» состояние и будет иметь заранее заданное значение ID\_Address в регистре хранения 0x00 (0x1A). Регистры DI\_1\_counter, DI\_2\_counter, DI\_3\_counter, DI\_4\_counter и регистры фиксируемых тревог будут установлены в 0, так что будьте осторожны.

## 7.4 ШИРОКОВЕЩАНИЕ

Широковещание не поддерживается устройством GPIO.

## 7.5 СТОРОЖЕВОЙ ТАЙМЕР

Сторожевой таймер – это таймер, реализованный в GPIO для проверки работоспособности связей в шине. В соответствии с этими описаниями требуется задание времени, после которого происходит срабатывание сторожевого таймера, и задачи, выполняемой в этом случае.

Если GPIO теряет связь на время, превышающее это значение времени watchdog\_time, то самый старший бит в регистре фиксируемых тревог в регистре хранения 0x20 и во входном регистре 0x01 мгновенных тревог будет установлен в 1.

Примечание: Если watchdog\_time меньше 30 секунд, то при первом включении GPIO этот таймер сработает через 30 секунд из соображений защиты.

### 7.5.1. Время СТОРОЖЕВОГО ТАЙМЕРА

Время срабатывания сторожевого таймера задано в регистре хранения 0x10. Это время устанавливается в секундах. Значение 0 означает выключение сторожевого таймера, и это значение принято по умолчанию. При каждом срабатывании сторожевого таймера его счетчик WDT\_counter увеличивается, как это описано в разделе 9.3.

Для включения сторожевого таймера установите watchdog\_time отличным от 0.

Пример: Установить время сторожевого таймера равным 30 секундам:

Передаваемое сообщение: 1A 10 00 10 00 01 02 00 1E 9A 38

Где:

- 1A это адрес ведомого.
- 10 это используемая функция. Запись нескольких регистров.
- 00 10 это адрес регистра хранения для записи.
- 00 01 это количество регистров, подлежащих записи, 1 в данном случае.
- 02 это количество байтов данных, подлежащих передаче.
- 00 1E это посылаемое значение, 30 в десятичной системе.
- 9A 38 это CRC.

Теперь watchdog\_time установлено равным 30 секундам. Теперь, если два правильно составленных сообщения считываются с промежутком менее 30 секунд, даже если они не адресованы GPIO, то сторожевой таймер не срабатывает. В противном случае он срабатывает.

Чтобы узнать значение watchdog\_time, необходимо прочесть регистры хранения.

Получаемое сообщение: 1A 03 02 00 1E 5C 4E

- 1A это адрес ведомого.
- 03 это используемая функция. Чтение регистров хранения.
- 02 это количество байтов данных, подлежащих чтению.
- 00 1E это конфигурируемое время (30 в десятичной системе).
- 5C 4E это CRC.

### 7.5.2 Конфигурация СТОРОЖЕВОГО ТАЙМЕРА

Расположенная в регистре хранения 0x11, конфигурация устанавливает, какое состояние имеет место при срабатывании сторожевого таймера.

**Старший байт:**

**ЕСЛИ РАВЕН 0, ТО ПРИ СРАБАТЫВАНИИ СТОРОЖЕВОГО ТАЙМЕРА GPIO ПЕРЕХОДИТ В СОСТОЯНИЕ «СТОРОЖЕВОЙ ТАЙМЕР».**

**ЕСЛИ ОН ОТЛИЧАЕТСЯ ОТ 0, ТО ПРИ СРАБАТЫВАНИИ СТОРОЖЕВОГО ТАЙМЕРА УСТРОЙСТВО ВОЗВРАЩАЕТСЯ В СОСТОЯНИЕ «СТАРТ».**

**Младший байт:** не используется в GPIO.

Пример: Конфигурирование сторожевого таймера для входа в состояние «Сторожевой таймер» при срабатывании таймера.

Передаваемое сообщение: 1A 10 00 11 00 01 02 00 00 1B E1

Где: 1A это адрес ведомого.

10 это используемая функция. Запись нескольких регистров.

00 11 это адрес первого регистра для записи.

00 01 это количество регистров, подлежащих записи, 1 в данном случае.

02 это количество байтов данных, подлежащих передаче.

00 00

**Старший байт.** Устройство перейдет состояние «Сторожевой таймер» при срабатывании таймера.

**Младший байт.** Не используется.

1B E1 это CRC.

### 7.5.3 Состояние реле WDT

Заранее заданное состояние реле сторожевого таймера в регистре хранения 0x14 определяет, будут ли четыре реле размыкаться или замыкаться (соответственно) при срабатывании сторожевого таймера.

Четыре реле будут размыкаться или замыкаться в соответствии с двоичным значением, посылаемым в этот регистр, от реле самого младшего бита до четвертого бита в нарастающем порядке.

Пример: установить WDT\_relay\_state для следующего порядка работы: первое и третье реле ВКЛЮЧЕНЫ, а остальные ВЫКЛЮЧЕНЫ (RL4=0; RL3=1; RL2=0; RL1=1).

Передаваемое сообщение: 1A 10 00 14 00 01 02 00 05 DB B7

Где:

1A это адрес ведомого.

10 это используемая функция. Запись нескольких регистров.

00 14 это адрес первого регистра для записи.

00 01 это количество записываемых байтов.

02 это количество байтов, подлежащих передаче.

00 05 это двоичное число, эквивалентное размыканию только первого и третьего реле.

DB B7 это CRC.

**ПРИМЕЧАНИЕ:** Помните, что функция сторожевого таймера реализована для обеспечения безопасности.

Однако ее активация зависит от критериев установщика/специалиста по интеграции. Всегда помните о последствиях ее реализации. Используйте это свойство под свою ответственность.

Watchdog\_time, Watchdog\_config и WDT\_relay\_state должны конфигурироваться совместно.

## 8. Цифровые входы

Входной регистр 0x20 может получать значение дискретного сигнала, такого как сухой контакт или влажный контакт с оптической развязкой. Задние фронты этого сигнала могут фильтроваться (описано в разделах 9.1 «Счетчики цифровых входов» и 8.1 «Конфигурация фильтра цифрового входа», соответственно). На рисунке 6 показан пример использования считывания импульсов двух устройств.

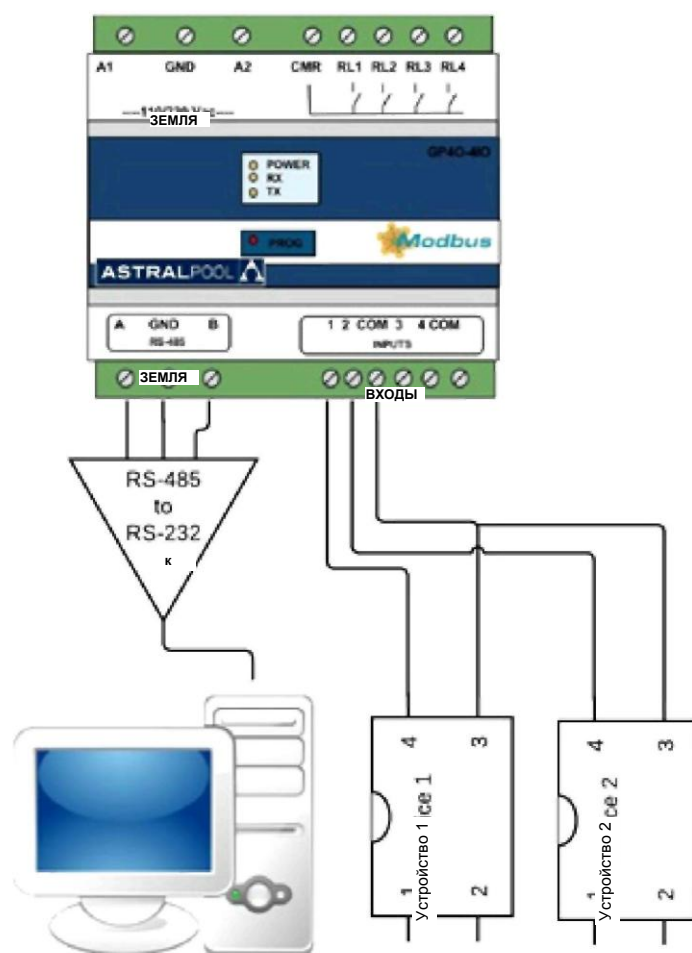


Рисунок 6. Цифровые входы в GPIO

### 8.1 Конфигурация фильтра цифрового входа

Некоторые сигналы могут иметь ложное поведение и создавать различные проблемы и несогласующиеся измерения. Эти соединения могут иметь случайные пики, создаваемые физическими отражениями, и счетчики могут хранить более одного повышения на каждый задний фронт, как показано на верхнем графике рисунка 7.

Назначение этого фильтра – избежать этих дополнительных задних фронтов и достичь одного повышения на каждый импульс, как показано на нижнем графике рисунка 7.

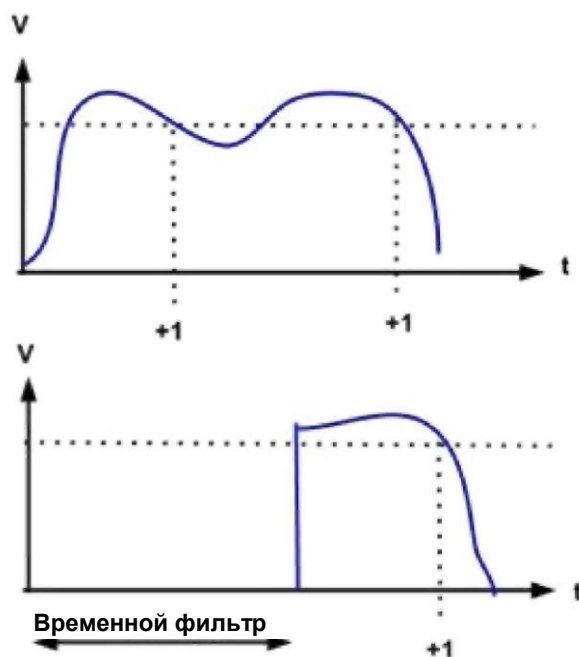


Рисунок 7. Ложный сигнал без/с цифровым фильтром.

Эта проблема может быть решена настройкой конфигурации фильтра цифрового входа (расположенной в регистре хранения 0x12) путем задания времени, в течение которого вход игнорирует сигнал, а после истечения этого времени сигнал будет считываться правильно.

Это время должно быть больше, если сигнал имеет больше пиков, и уменьшаться или быть равным нулю, если сигнал имеет меньше пиков или вообще не имеет их. Параметрами фильтра будут:

**0x12 Старший байт** Время, необходимое, чтобы избежать плохих измерений.

- 0 Без фильтра
- 1 Временной фильтр, равный 1 миллисекунде
- 2 Временной фильтр, равный 10 миллисекундам
- 3 Временной фильтр, равный 100 миллисекундам
- 4 Временной фильтр, равный 500 миллисекундам

**0x12 Младший байт** Вход или входы, подлежащие фильтрации.

Пример: настройка входов 1, 3 и 4 фильтром с временем 500 миллисекунд

Передаваемое сообщение: 1A 10 00 12 00 01 02 04 0D D8 D7

Где

- 1A это адрес ведомого.
- 10 это используемая функция. Запись нескольких регистров.
- 00 12 это адрес первого регистра для записи.
- 00 01 это количество записываемых регистров.
- 02 это количество байтов, подлежащих записи.
- 04 0D

**Старший байт** – это конфигурация фильтра 0000 0100, означающая время 500 миллисекунд.

**Младший байт** – это конфигурация фильтра 0000 1101, относящаяся к входам 1, 3 и 4.

D8 D7 это CRC.

Получаемое сообщение: 1A 10 00 12 00 01 A2 27

## 8.2 Чтение влажных контактов

Влажные контакты имеют источник питания, имеющийся в оборудовании; если требуется считывать такой контакт, то нужен **оптоизолятор**, чтобы обеспечить изоляцию между входами и влажными контактами;

подключение контактов должно быть организовано в соответствии с рисунком 9.

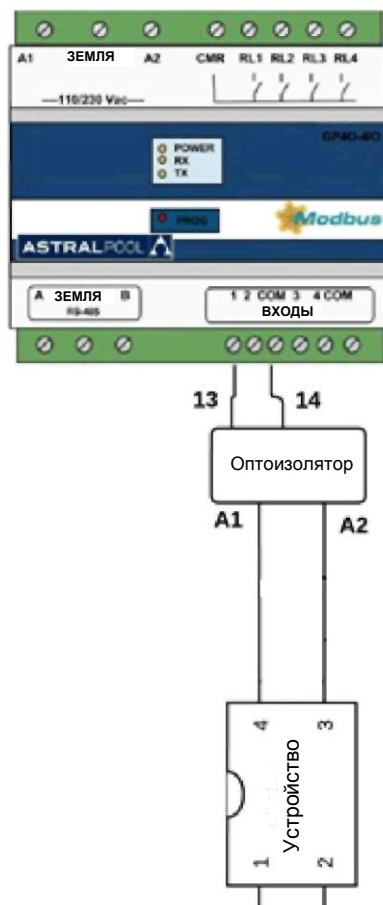


Рисунок 8. Считывание влажных контактов с помощью входов GPIO

Если требуется дополнительная информация об этом типе соединений, то, пожалуйста, обратитесь к справочному листку любого оптоизолятора.

## 9 Счетчики

GPIO имеет три вида счетчиков, каждый из которых предназначен для подсчета различных параметров, связанных с поведением GPIO.

### 9.1 Счетчики цифровых входов

Эти счетчики расположены во входных регистрах (от 0x03 до 0x06) и могут нарастать только при обнаружении входом заднего фронта (перехода от 1 в 0). Они имеют названия DI\_1\_counter, DI\_2\_counter, DI\_3\_counter и DI\_4\_counter.

Помните, что если GPIO перепрограммирован в соответствии с процедурой, описанной в разделе 7.3.3 «Установка заводской конфигурации с помощью переключателя PROG», то эти четыре входных регистра будут сброшены в 0.

Пример: считывать счетчики цифровых входов в качестве входных регистров. Значения, полученные после сброса устройства в заводскую конфигурацию.

Передача: 1A 04 00 03 00 04 02 22

- 1A это адрес ведомого.
- 04 это используемая функция. Чтение входных регистров.
- 00 03 это адрес регистра хранения, подлежащего чтению.
- 00 04 это количество байтов, подлежащих чтению.
- 02 22 это CRC.

Получаемое сообщение: 1A 04 08 00 00 00 00 00 00 00 55 E9

### 9.2 Счетчик питания

Расположен в регистре хранения 0x30 и

наращивается при каждом включении питания GPIO.

Пример: проверить значение power\_counter перед и после сброса устройства в заводские настройки.

Передаваемое сообщение (перед сбросом): 1A 03 00 30 00 01 87 EE

Где

- 1A это адрес ведомого.
- 03 это используемая функция. Чтение регистров хранения.
- 00 30 это адрес регистра хранения, подлежащего чтению.
- 00 01 это количество байтов, подлежащих чтению.
- 87 EE это CRC.

Получаемое сообщение: 1A 03 02 **00 3F** 9C 56

Передаваемое сообщение (после сброса): 1A 03 00 30 00 01 87 EE

Получаемое сообщение: 1A 03 02 **00 40** DD B6

Где: 1A это адрес ведомого.

- 03 это используемая функция. Запись нескольких регистров.
- 02 это количество байтов, подлежащих чтению.
- 00 40 это полученные байты
- DD B6 это CRC.

### 9.3 Счетчик WTD

Расположен в регистре хранения 0x31 и наращивается при каждом срабатывании сторожевого таймера.

Пример: проверка счетчика WDT до и после срабатывания сторожевого таймера.

Передаваемое сообщение (до срабатывания сторожевого таймера): 1A 03 00 31 00 01 D6 2E

Получаемое сообщение: 1A 03 02 **44 D3** AE DB

Где

- 1A это адрес ведомого.
- 03 это используемая функция, чтение регистров хранения.
- 02 это количество получаемых байтов.
- 44 D3 это значение счетчика WDT.
- AE DB это CRC.

Количество раз, когда GPIO входит в состояние «Сторожевой таймер» равно 44D3 в шестнадцатеричной системе.

Передаваемое сообщение (после срабатывания сторожевого таймера): 1A 03 00 31 00 01 D6 2E

Получаемое сообщение: 1A 03 02 **44 D4** EF 19

Где

- 1A это адрес ведомого.
- 03 это используемая функция, чтение регистров хранения.
- 02 это количество получаемых байтов.
- 44 D4 это значение счетчика WDT.
- EF 19 это CRC.

## 10. Режимы работы

### 10.1. Основной режим

В этом разделе предполагается, что было установлено успешно соединение с GPIO, и поэтому адрес, настройки связи и остальные конфигурации уже были установлены.

#### 10.1.1. Заранее заданное начальное состояние реле

Это расположение реле в GPIO при ВКЛЮЧЕНИИ питания (расположенное во входном регистре 0x15 и катушках от 150 до 153); на рисунке 9 показано расположение этих регистров хранения и катушек в устройстве.

Младший байт 0x15 регистра хранения начального состояния реле



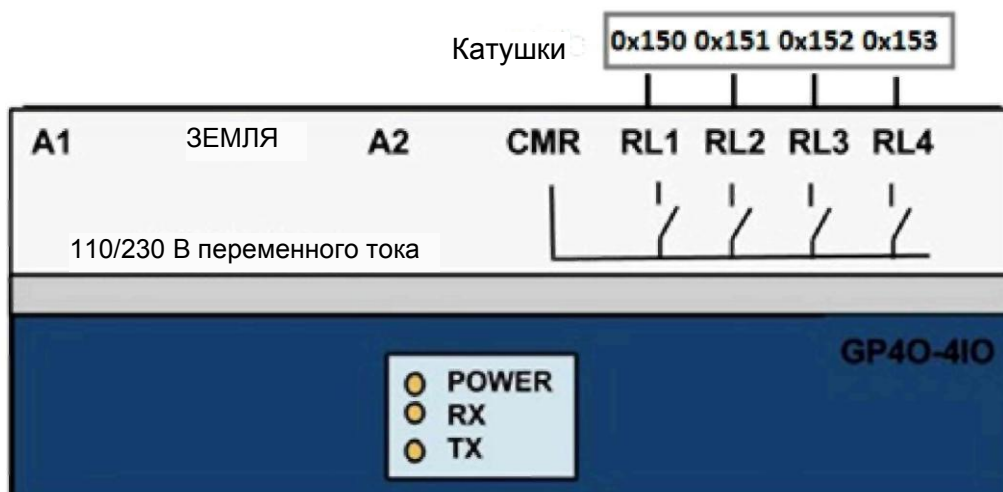


Рисунок 9. Начальное состояние реле

Пример: установить начальное состояние реле равным 0.

Передаваемое сообщение: 1A 10 00 15 00 01 02 00 00 1A 65

Где:

- 1A это адрес ведомого (реальный ID\_Address).
- 10 это используемая функция, запись нескольких регистров.
- 00 15 это адрес регистра хранения, подлежащего записи.
- 00 01 это количество регистров, подлежащих записи.
- 02 это количество байтов данных, подлежащих передаче.
- 00 00

**Старший байт:** не используется.

**Младший байт:** четыре младших бита относятся к четырем реле в нарастающем порядке.

1A 65 это CRC.

Получаемое сообщение: 1A 10 00 15 00 01 13 E6

### 10.1.2. Проверка текущего состояния

Состояние, в котором в настоящее время находится GPIO, доступно через входной регистр состояния 0x00. Этот регистр имеет разные значения для младшего и старшего байтов.

Младший бит используется для показа наличия ошибки и устанавливается в 1, если имеется ошибка.

Подробная информация об обнаруженной ошибке/ошибках может быть зарегистрирована во входном регистре тревоги 0x01 или регистре хранения фиксируемых тревог 0x20 (описанных в разделах 10.1.5 «Проверка регистра мгновенных тревог» и 10.1.4 «Проверка регистра фиксируемых тревог», соответственно).

Была реализована только тревога сторожевого таймера, которая означает, что имеется ошибка, так как сработал сторожевой таймер.

Коды для различных состояний, реализованных в старшем байте, показаны в следующей таблице:

Старший байт	Младший байт
00 Старт	00 нет ошибок
01 Сторожевой таймер	01 ошибка
02 Запрос	

Примеры кодирования (сначала старший байт):

- 0x000 Состояние «Старт» и нет ошибки.
- 0x001 Состояние «Старт» и ошибка из-за сторожевого таймера.
- 0x100 Состояние «Запрос» и нет ошибки.
- 0x201 Состояние «Сторожевой таймер» и ошибка из-за сторожевого таймера.

Пример: Чтение текущего состояния в GPIO.

Передаваемое сообщение: 1A 04 00 00 00 01 32 21

Где:

1A это адрес ведомого.  
 04 это используемая функция. Чтение входных регистров.  
 00 00 это адрес первого регистра, подлежащего чтению.  
 00 01 это количество регистров, подлежащих чтению.  
 32 21 это CRC.

Получаемое сообщение: 1A 04 02 **02 01** 1D 92, что указывает на состояние «Сторожевой таймер» с тревогой сторожевого таймера.

Где:

1A это адрес ведомого.  
 04 это используемая функция. Чтение входных регистров.  
 02 это полученные байты.  
 02 01

**Старший байт:** GPIO находится в состоянии «Сторожевой таймер».

**Младший байт:** GPIO имеет некоторую тревогу (сторожевой таймер).

1D 92 это CRC.

### 10.1.3. Запрос изменения реле

Запрос изменения реле может быть послан в GPIO через control\_w0 (расположенный в регистре хранения 0x21).

Однако эти запросы могут также посылаются, используя режим битового адреса, начиная с катушки 0x210. Оба способа изменения реле не должны смешиваться, чтобы избежать неправильного понимания; на рисунке 10 показано расположение этих регистров хранения и катушек в устройстве.

Младший байт регистра хранения CONTROL W0 (0x21)

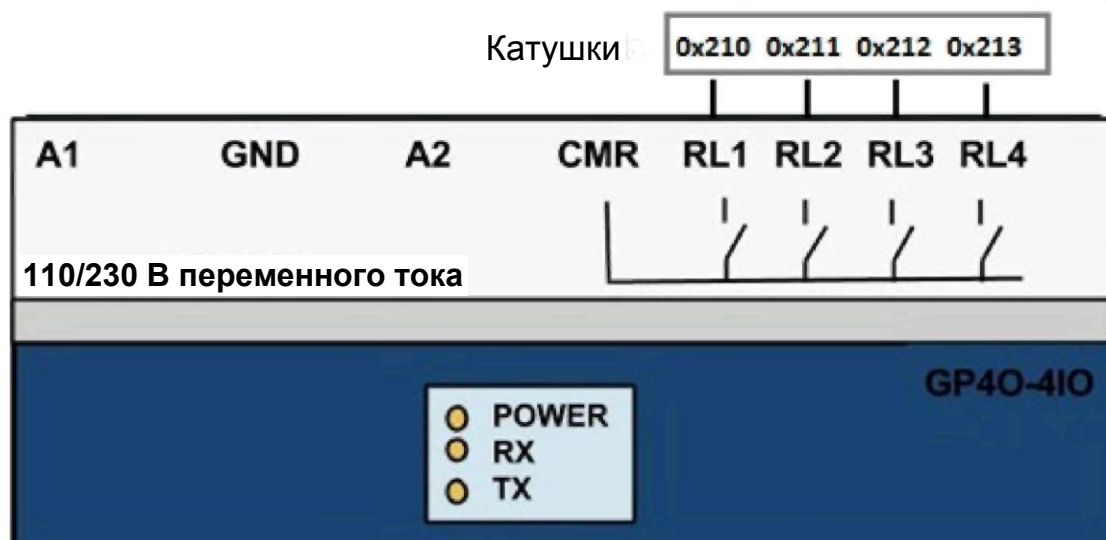


Рисунок 10. Таблица расположения выходных реле

Пример: активирование реле 1 и 3

Передаваемое сообщение: 1A 10 00 21 00 01 02 00 05 DE 12

Где:

1A это адрес ведомого.  
 10 это используемая функция. Запись нескольких регистров.  
 00 21 это адрес регистра хранения, подлежащего записи.  
 00 01 это количество регистров, подлежащих записи.  
 02 это количество байтов данных, подлежащих передаче.  
 00 05

**Старший байт:** не используется.

**Младший байт:** это двоичное число, эквивалентное размыканию только первого и третьего реле.  
1A 65 это CRC.

Получаемое сообщение: 1A 10 00 21 00 01 52 28

#### 10.1.4. Проверка регистра фиксируемых тревог

Можно проверить, какая тревога была активирована после последнего включения питания или последнего сброса тревог. Были реализованы только тревоги сторожевого таймера.

Для проверки этого должен быть прочитан регистр хранения 0x20, причем биты от 0 до 14 не используются, но могут считываться, а бит 15 является тревогой сторожевого таймера; это можно также считывать в качестве катушек (расположенных от катушки 0x200 до 0x20F).

Значение 1 указывает на то, что тревога была активирована, а 0 означает отсутствие активизации.

При пропадании питания все тревоги будут сброшены в 0. Эти значения тревоги могут быть также принудительно установлены.

Пример: чтение фиксируемых тревог с помощью регистров хранения, когда была активирована тревога сторожевого таймера.

Передаваемое сообщение: 1A 03 00 20 00 01 86 2B (чтение регистра хранения 0x20).

Получаемое сообщение: 1A 03 02 80 00 BD 86

Где:

1A это адрес ведомого.  
03 это используемая функция. Чтение регистров хранения.  
02 это количество байтов данных, подлежащих чтению.  
80 00

**Старший байт:** Самый старший бит указывает ошибку сторожевого таймера.

**Младший байт.** Не используется.

BD 86 это CRC.

Пример: чтение фиксируемых тревог с помощью катушек, когда была активирована тревога сторожевого таймера.

Передаваемое сообщение: 1A 01 02 0F 00 01 CF 9A (чтение катушки 0x20F).

Получаемое сообщение: 1A 01 01 01 96 AC

Где:

1A это адрес ведомого.  
01 это используемая функция. Чтение катушек.  
01 это количество катушек, подлежащих чтению.  
01 указывает, что тревога была активирована.  
96 AC это CRC.

Пример: сбросить все тревоги записью 00 00 в регистре хранения 0x20.

Передаваемое сообщение: 1A 10 00 20 00 01 02 00 00 1F C0

Где:

1A это адрес ведомого.  
10 это используемая функция. Запись нескольких регистров.  
00 20 это адрес первого регистра хранения, подлежащего записи.  
00 01 это количество регистров хранения, подлежащих записи, 1 в данном случае.  
02 это количество байтов данных, подлежащих передаче.  
00 00 это посылаемые данные (0 для выключения тревоги).  
1F C0 это CRC.

Получаемое сообщение: 1A 10 00 20 00 01 03 E8

#### 10.1.5. Проверка регистра мгновенных тревог

Если ошибка существует, то самый младший бит во входном регистре состояния 0x00 (описанном в разделе

10.1.2 «Проверка текущего состояния») и самый старший бит Цифрового входа мгновенных тревог 0x01 установлены в 1.

Пример: Проверка регистра мгновенных тревог.

Передаваемое сообщение: 1A 04 00 01 00 01 63 E1

Где

- 1A это адрес ведомого.
- 04 это используемая функция. Чтение входных регистров.
- 00 01 это адрес первого регистра, подлежащего чтению.
- 00 01 это количество входных регистров, подлежащих записи.
- 63 E1 это CRC.

Получаемое сообщение: 1A 04 02 80 00 BC F2

Где:

- 1A это адрес ведомого.
- 04 это используемая функция. Чтение входных регистров.
- 02 это количество байтов, подлежащих чтению.
- 80 00 это значение считанных входных регистров, только самый старший бит установлен в 1.
- BC F2 это CRC.

Расположение этого входного регистра ошибки 0x01 показано в разделе 11 «Карта основных регистров ModBus RTU», биты от 0 до 14 не используются, но могут считываться без ограничения, тогда как бит 15 показывает ошибку сторожевого таймера MODBUS.

## 11. КАРТА ОСНОВНЫХ РЕГИСТРОВ MODBUS RTU

Таблица, показанная в этом разделе, является нашей исключительной и оригинальной картой регистров с названиями функций и их адресами.

Кроме самой карты регистров имеется прямая связь между регистром хранения 0x21 и входным регистром 0x00. Тогда как один из них поддерживает действие запроса, второй из них устанавливает текущее исполняемое действие.

Имеется также прямая связь между регистром хранения фиксируемых тревог 0x20 и входным регистром мгновенных тревог 0x01. Тогда как регистр хранения 0x20 является итогом фиксируемых тревог, входной регистр 0x01 является текущей тревогой. Эта связь также применима к побитовому соответствию между регистрами. Была реализована только тревога сторожевого таймера, а это значит, что фиксируемые тревоги и мгновенные тревоги всегда соответствуют друг другу.

Для сброса тревог ошибок необходимо сбросить их из Регистра хранения 0x20, а не из Входного регистра 0x01; поэтому Входной регистр 0x01 будет сброшен, когда исчезнет текущая тревога ошибки. Для сброса всех тревог необходимо сбросить в 0 катушки от 0x200 до 0x20F.

Примечание: отключение электропитания сбросит все фиксируемые тревоги.

Название	Регистры хранения	Входные регистры	Катушки	Входы	Описание
ID_Address	0x00				Этот параметр является адресом GPIO; по умолчанию он установлен равным 0x1A. Если установка имеет несколько устройств, подключенных к шине, то необходимо изменить ID_Address устройства до подключения другого устройства.
					Этот параметр определяет Выбранную скорость обмена.
COM_Setup	0x01				Имеются 6 возможных конфигураций для выбора: Заводская настройка 0. 9600, 8E1 Поддерживаемые значения: 1. 19200, 8E1 2. 9600, 8N2 3. 19200, 8N2 4. 9600, 8N1 5. 19200, 8N1
ID_Manufacturer_hi	0x02				Этот параметр указывает старший байт, который представляет Код производителя.
ID_Manufacturer_lo	0x03				Этот параметр указывает младший байт, который представляет Код производителя.
ID_Product_code_hi	0x04				Этот параметр указывает старший байт, который представляет Код изделия.
ID_Product_code_lo	0x05				Этот параметр указывает младший байт, который представляет Код изделия.
Зарезервировано	0x06				Зарезервировано.
HW_Version	0x07				Этот параметр указывает Аппаратную версию GPIO.
SW_Version	0x08				Этот параметр указывает Программную версию GPIO.
MODEL_Serie_hi	0x09				Этот параметр указывает старший байт Серийного номера.
MODEL_Serie_lo	0x0A				Этот параметр указывает младший байт Серийного номера.
MODEL_Production_hi	0x0B				Этот параметр указывает старший байт производственной партии.
MODEL_Production_low	0x0C				Этот параметр указывает младший байт производственной партии.
Watchdog_time	0x10				Этот параметр задает время, выраженное в секундах, которое может пройти между последней связью и активацией сторожевого таймера. Если оно установлено в 0, то сторожевой таймер отключен. Диапазон: 0-65535
Watchdog_config	0x11 младший байт старший байт				Этот параметр определяет, как будет активизироваться сторожевой таймер при отказе связи. Младший байт – это состояние при срабатывании сторожевого таймера (не используется). Старший байт Если он равен 0 при активном состоянии «Сторожевой таймер», то происходит переход в состояние «Сторожевой таймер». Если он не равен 0 при активном состоянии «Сторожевой таймер», то происходит сброс GPIO.

DigInputFilterConfig	0x12 биты 0...3 биты 7...10				Этот параметр задает время фильтра для игнорирования отражений и фильтрации входов. Двоичное значение Младшего байта – это вход или входы, подлежащие фильтрации. Старший байт 0 Без фильтра. 1 Фильтр со временем 1 миллисекунда. 2 Фильтр со временем 10 миллисекунд. 3 Фильтр со временем 100 миллисекунд. 4 Фильтр со временем 500 миллисекунд.
wdt_relay_state	0x14 биты 0...3		0x140 0x141 0x142		Этот регистр устанавливает положение реле, когда GPIO входит в состояние «Сторожевой таймер». Младший байт: В двоичном коде: состояние, желаемое для четырех реле. RL1 RL2
			0x143		RL3 RL4
Положение реле в состоянии «Старт»	0x15 биты 0...3		0x150 0x151 0x152 0x153		Этот регистр устанавливает положение реле, когда GPIO входит в состояние «Сторожевой таймер». Младший байт: В двоичном коде: 0 = реле замкнуто, 1 = реле разомкнуто.  RL1 RL2 RL3 RL4
Фиксируемая тревога Сторожевой таймер	0x20 биты 0...14 бит 15		От 0x200 до 0x20E  0x20F		Этот параметр указывает, какая тревога была активизирована после ВКЛЮЧЕНИЯ питания GPIO. Самый старший бит: 1 = сторожевой таймер сработал, 0 = сторожевой таймер не сработал.
control_w0	0x21 биты 0...3  биты 4..15		0x210 0x211 0x212 0x213 От 0x214 до 0x21F		Младший байт: В двоичном коде.  RL1. 0 = реле замкнуто, 1 = реле разомкнуто RL2. 0 = реле замкнуто, 1 = реле разомкнуто RL3. 0 = реле замкнуто, 1 = реле разомкнуто RL4. 0 = реле замкнуто, 1 = реле разомкнуто не используется.
Счетчики power_counter WDT_counter	0x30 0x31				Счетчик увеличивается, когда GPIO теряет электропитание (запись заблокирована). Счетчик увеличивается, когда GPIO входит в состояние «Сторожевой таймер» (запись заблокирована).
DI_1_counter DI_2_counter DI_3_counter DI_4_counter		0x03 0x04 0x05 0x06			Этот счетчик увеличивается, когда на входе обнаруживается задний фронт сигнала.
Состояние Тревога  ВКЛ  0: Состояние «Старт»		0x00 младший байт  бит 0 старший байт		0x000	Младший байт: Указывает тревогу, 1 = Тревога сторожевого таймера. 0 = Нет тревоги сторожевого таймера.



1: Состояние «Запрос» 2: Состояние «Сторожевой таймер»		биты 8...11		0x008... 0x00B	Старший байт: Указывает реальное состояние.
Мгновенная тревога		0x01 биты 1...14			Этот параметр указывает мгновенные тревоги при выполнении чтения.
Сторожевой таймер		бит 15		0x01F	Самый старший бит: 1 = сторожевой таймер сработал, 0 = сторожевой таймер не сработал.
Цифровые входы		0x02 биты 0...3		0x020 0x021 0x022 0x023	Этот регистр задает вход GPIO, который может считывать импульсы. Вход 1 Вход 2 Вход 3 Вход 4

## 12. ВЕРСИЯ ИЗДЕЛИЯ

Руководство v.1.04: Информация в этом руководстве описывает поведение Аппаратной версии 1.02 и Программной версии 1.17.