

## Описание протокола Modbus в рамках системы «Умный дом INSYTE» Версия 2.0

Протокол предполагает одно активное устройство в линии (контроллер SPIDER), которое может обращаться к нескольким пассивным устройствам по уникальному адресу. Синтаксис команд протокола позволяет адресовать 254 устройства, соединенных в линию.

Возможно применение протокола Modbus в дуплексных и полудуплексных линиях связи. Физическим уровнем протокола Modbus, как правило, является линия стандарта RS422/RS485, однако при соединении точка-точка тот же формат команд может быть использован на любом последовательном асинхронном физическом интерфейсе, в том числе RS232. Возможно применение протокола Modbus в сетевой среде поверх транспортных протоколов UDP/TCP и IPX/SPX. Протокол Modbus подразумевает наличие в линии только одного ведущего устройства (master) и множества (возможно и одно) подчиненных устройств (slave). Инициатива проведения обмена всегда исходит от ведущего устройства. Ведомые устройства прослушивают линию связи. Мастер подает запрос (посылка, последовательность байт) в линию и переходит в состояние прослушивания линии связи. Ведомое устройство отвечает на запрос, пришедший в его адрес.

Окончание ответной посылки мастер определяет, измеряя интервал времени между окончанием приема предыдущего байта и началом приема следующего. Если этот интервал превысил время, необходимое для приема двух байт на заданной скорости передачи, прием кадра ответа считается завершенным.

Таблица 1. Кадр посылки Modbus

поле кадра	Длина в байтах
Адрес подчиненного устройства	1
Номер функции	1
Данные	$N < 254$
контрольная сумма	2

\*адрес подчиненного устройства - первое однобайтное поле кадра. Оно содержит адрес подчиненного устройства, к которому адресован запрос. Подчиненные устройства отвечают только на запросы, поступившие в их адрес. Ответ также начинается с адреса отвечающего устройства. Может изменяться от 1 до 254;

\*номер функции - это следующее однобайтное поле кадра. Оно говорит подчиненному устройству, какие данные или выполнение какого действия требует от него ведущее устройство;

\*данные - поле содержит информацию необходимую подчиненному устройству для выполнения заданной мастером функции или содержит данные, передаваемые подчиненным устройством в ответ на запрос ведущего. Длина и формат поля зависит от номера функции;

\*контрольная сумма - заключительное двухбайтное поле кадра, содержащее циклическую контрольную сумму CRC-16 всех предыдущих полей кадра. Контрольная сумма завершает кадры запроса и ответа.

Таблица 2. Характеристики порта связи

Наименование	Характеристики
Стандарт передачи данных	RS 485
Протокол передачи данных	MODBUS-RTU
Формат данных	Стартовый бит; 8 бит данных; стоповый бит; бит контроля чётности (по умолчанию без контроля чётности)
Скорость передачи данных	38400 бит/сек

Протокол позволяет производить чтение и запись данных в регистры контроллеров, которым выделены пространства адресов.

Modbus-контроллеры могут интерпретировать свои данные, используя четыре типа параметров, которым выделены пространства адресов. Соответствие адресов и контролируемых параметров обычно указывается в технической документации на контроллер. На

чтение/изменение значений параметров каждого типа в протоколе существуют соответствующие команды.

Таблица 3. Типы данных MODBUS

Тип параметра	Тип величины	Формат	Адреса	Возможные операции	Смещение регистров
Coils	Дискретные	1 бит	00001-09999	чтение/запись	0
Discret Inputs	Дискретные	1 бит	10001-19999	чтение	10000
Input Registers	Аналоговые	16 бит	30001-39999	чтение	30000
Holding Registers	Аналоговые	16 бит	40001-49999	чтение/запись	40000

Для определения типа регистров в конфигураторе нажимаем правой кнопкой на устройство и выбираем «редактировать тип».

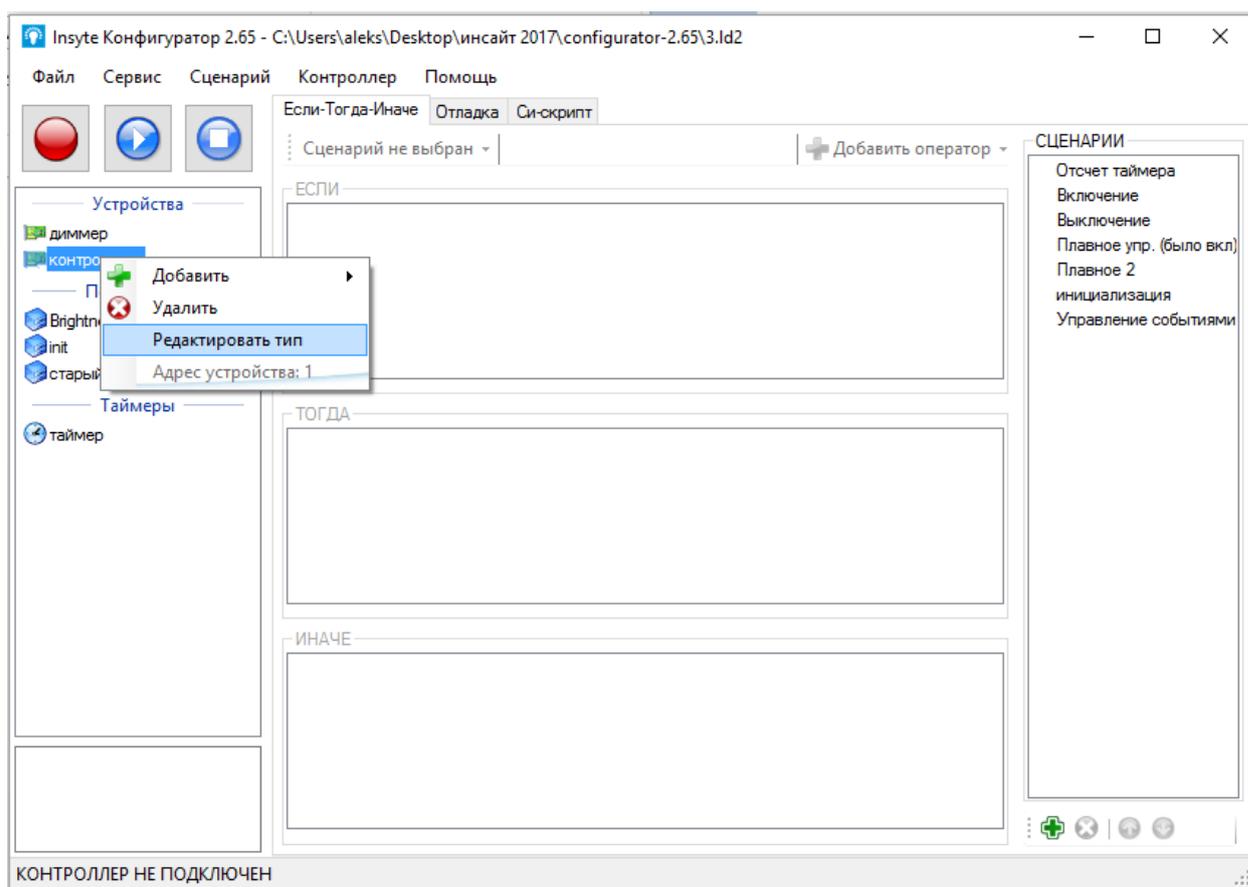


Рисунок 1 – Фрагмент программы

Далее заходим в поле реле 1, смотрим тип данных «Coils» (рисунок 2). Адрес 1 – адрес регистра 1 (данные в приложение заносим в смартфон). Данные регистры возможно считывать и записывать (то есть в приложение возможна функция запись, возможно включать нагрузку с реле выходного у контроллера), табл. 3.

Для интеграции в приложении необходимо произвести следующие настройки: адрес -1, регистр -1, функция: запись.

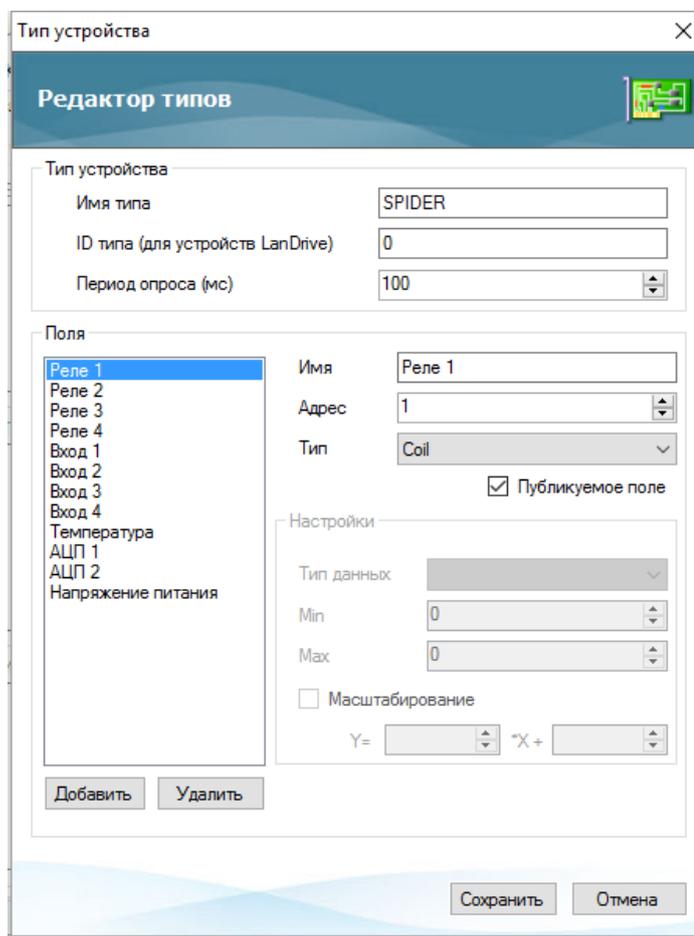


Рисунок 2 – Фрагмент программы реле 1

Далее заходим в поле реле 2, смотрим тип данных «Coils» (рисунок 3). Адрес 2 – 2 (данные в приложение заносим). Данные регистры возможно считывать и записывать, табл. 3.

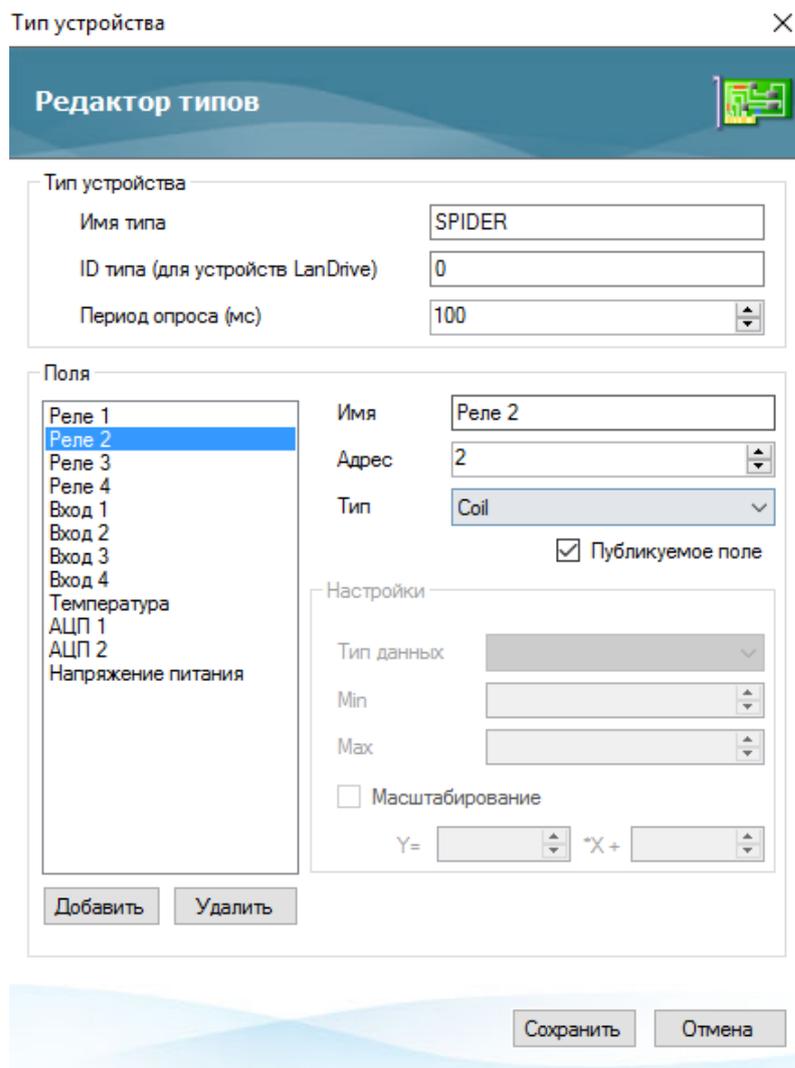


Рисунок 3 – Фрагмент программы реле -2

Аналогично реле -3 и реле -4.

Следующие поля – Вход 1, Вход 2, ....

Далее заходим в поле вход 1, смотрим тип данных «Discret Inputs» (данные можно только считывать, рисунок 4). Адрес 1, с учетом сдвига регистров записываем 10001 в мобильное приложение, функция – только чтение.

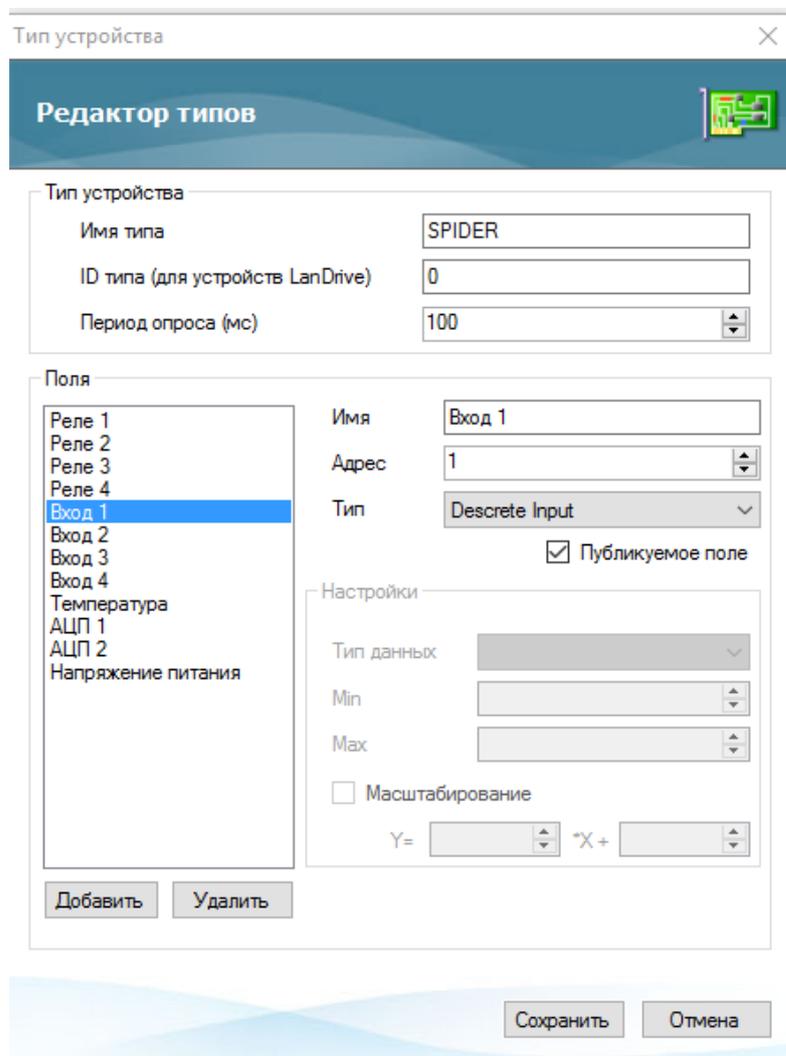


Рисунок 4 – Фрагмент программы Вход -1

Аналогично Вход -2 адрес 10 002, вход 3 – 10 003, вход 4 – 10 004.

Далее заходим в поле АЦП 1, смотрим тип данных «Holding registers» (рисунок 2). Адрес 1 – 40001 (данные в приложение заносим в смартфон).

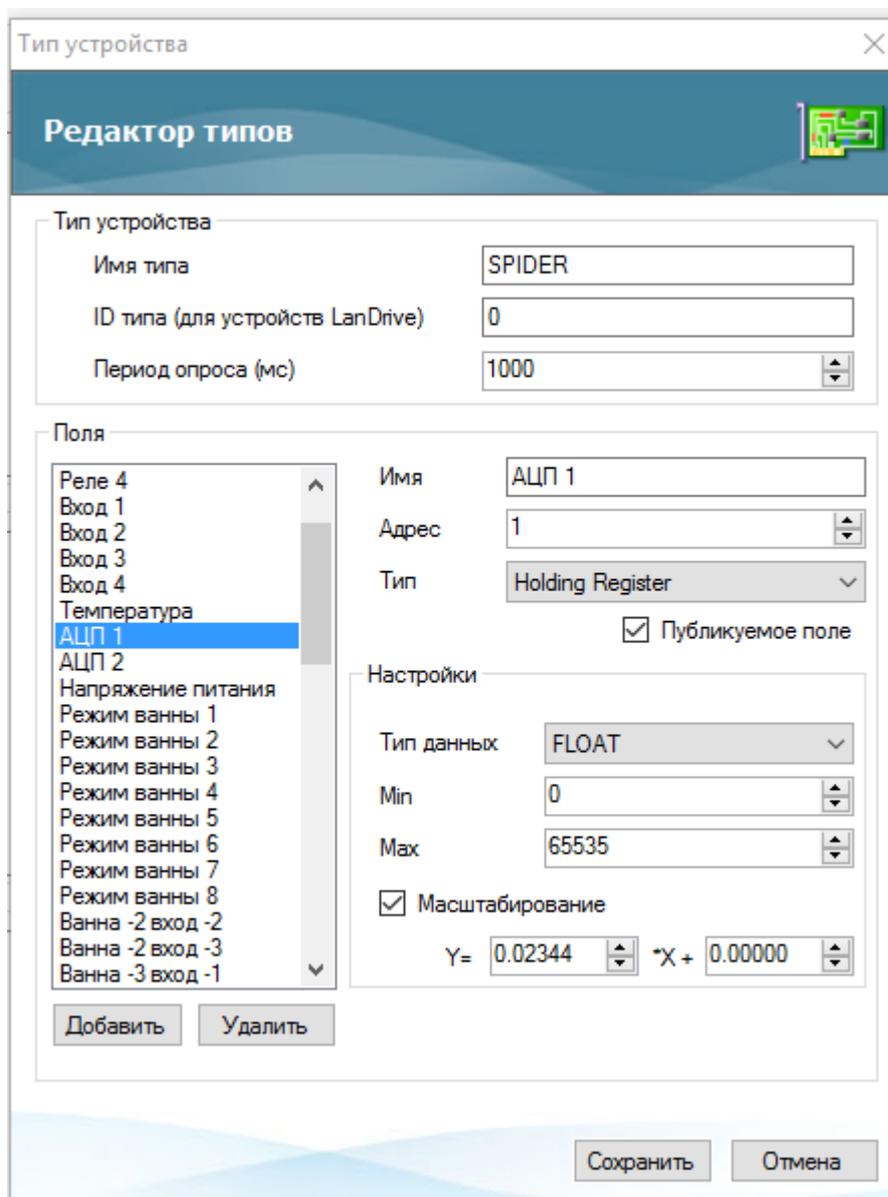


Рисунок 5 – Поле АЦП-1

Также для удобства можно создавать поля и указывать их тип регистров, наиболее часто встречаемый случай – создание дополнительных полей «Holding registers», главное – чтобы их адрес не совпадал с теми полями, которые уже есть в контроллере. Создадим поле «режим ванны 1» с адресом 7.

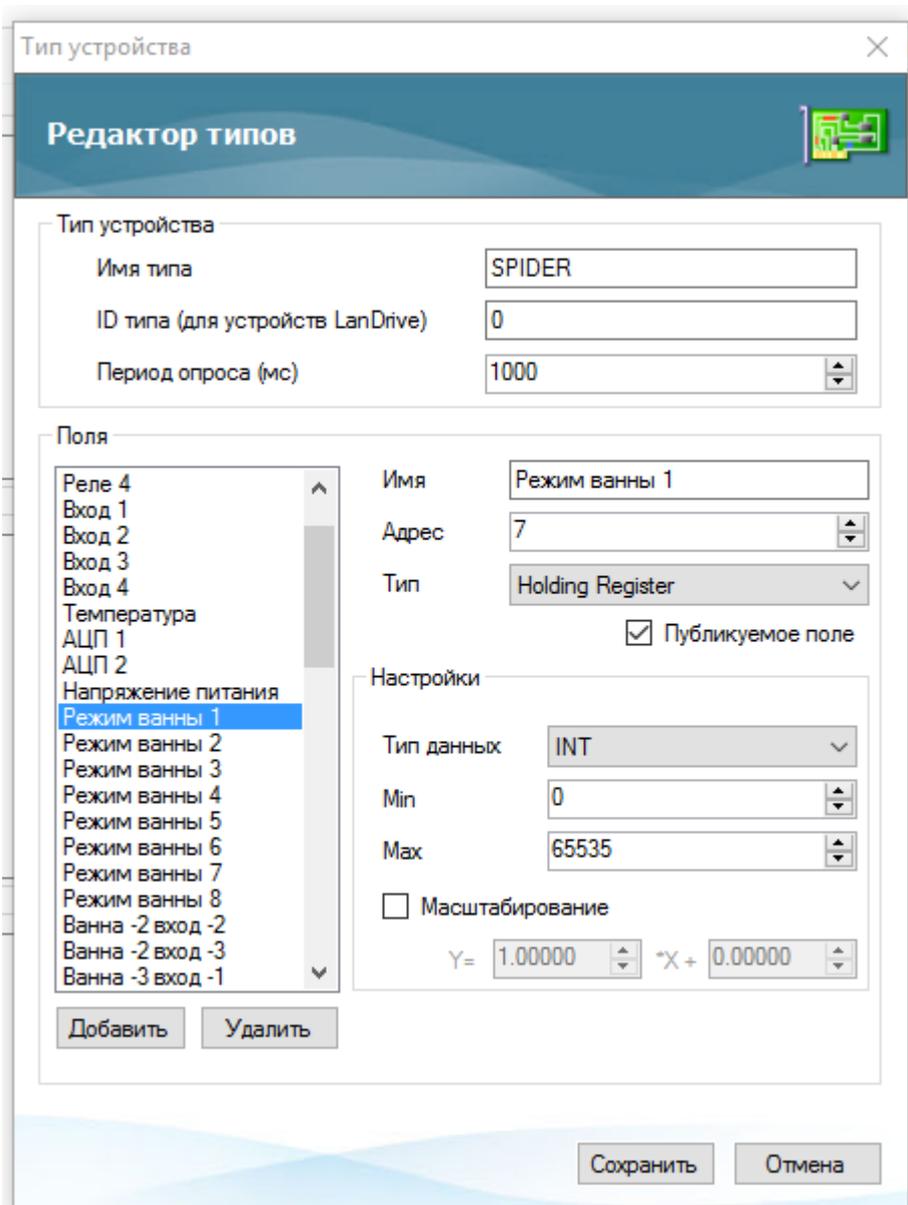


Рисунок 6 – Режим ванны

После создания это поле можно использовать в сценариях.

## Диммер

Нажимаем правой кнопкой мыши «редактировать тип».

**В мобильном приложении** соответствующие регистры можно найти в настройках, к примеру, заходим в освещение:

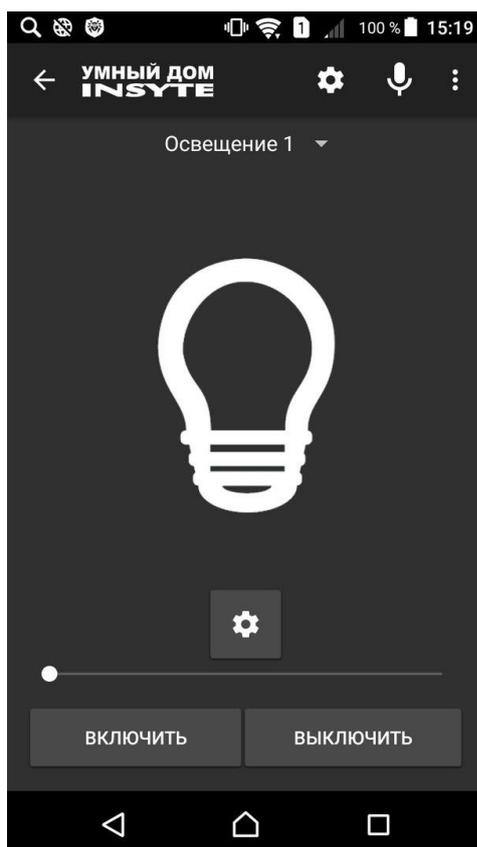


Рисунок 7 – Приложение

Далее в настройки – если нам нужно управлять яркостью: берем адрес из конфигуратора «Holding registers» адрес -2 (в приложение заносим: 40 002) и делаем функцию запись.

Также можно считывать состояние входов (Discret Inputs) в приложении.

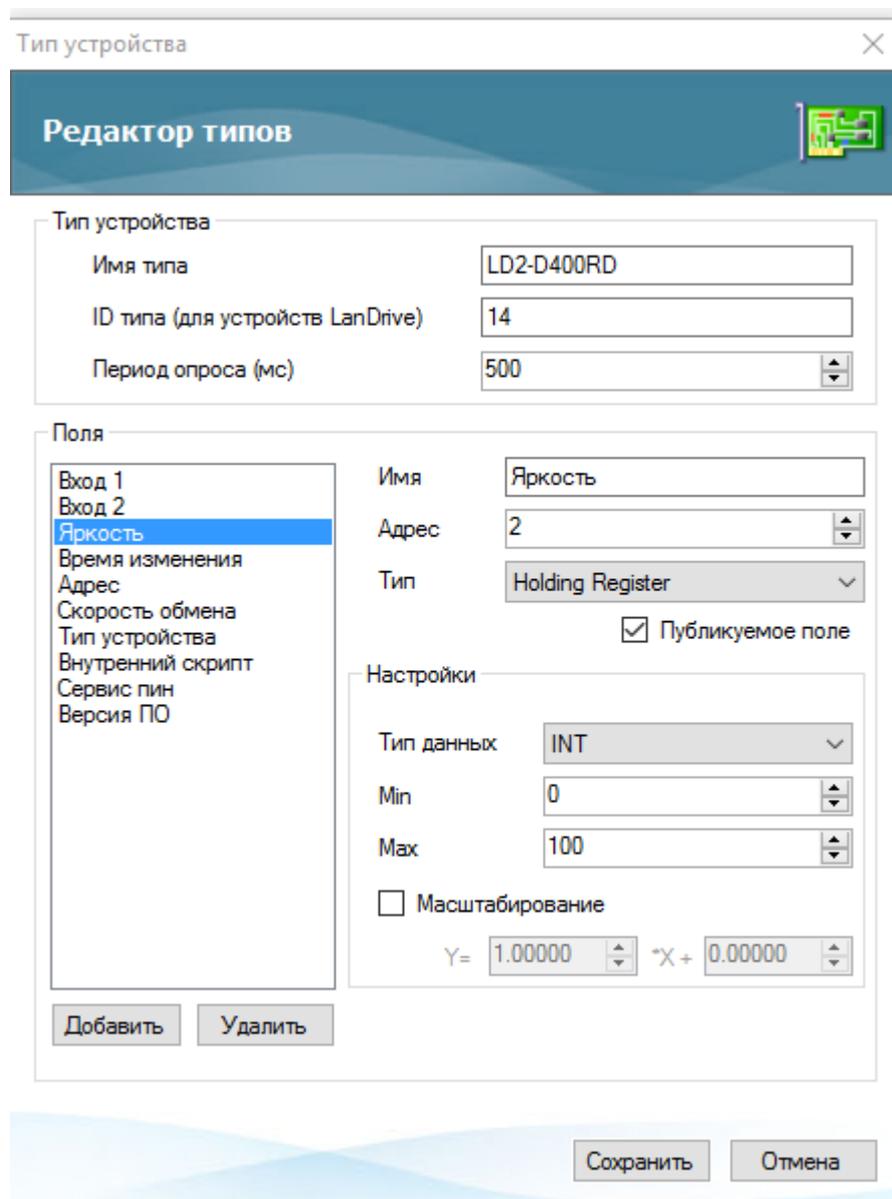


Рисунок 8 – Поле яркость. Диммер

Работа с Вход -1, Вход – 2 осуществляется аналогично контроллеру.

## Очень часто возникает следующая ситуация, когда нужно в программе создать переменные и считать значение с них в приложение или записать значение

К примеру задача: в конфигураторе необходимо создать целочисленную переменную, записать в нее значение, а затем его считывать в приложении (например, показания температуры с датчика).

Для целочисленных адрес регистра по формуле:

Adress (holding registr) = 41001 + (номер слота)\*2, в конфигураторе наводите на переменную и получаете слот.

Для примера получаем (слот 0):  $41001 + 0*2 = 41001$

Для данного примера получаем (слот 1):  $41001 + 1*2 = 41003$

Для примера получаем (слот 2):  $41001 + 2*2 = 41005$

Для примера получаем (слот 3):  $41001 + 3*2 = 41007$  и т.д.

Слоты для переменных создаются по порядку.

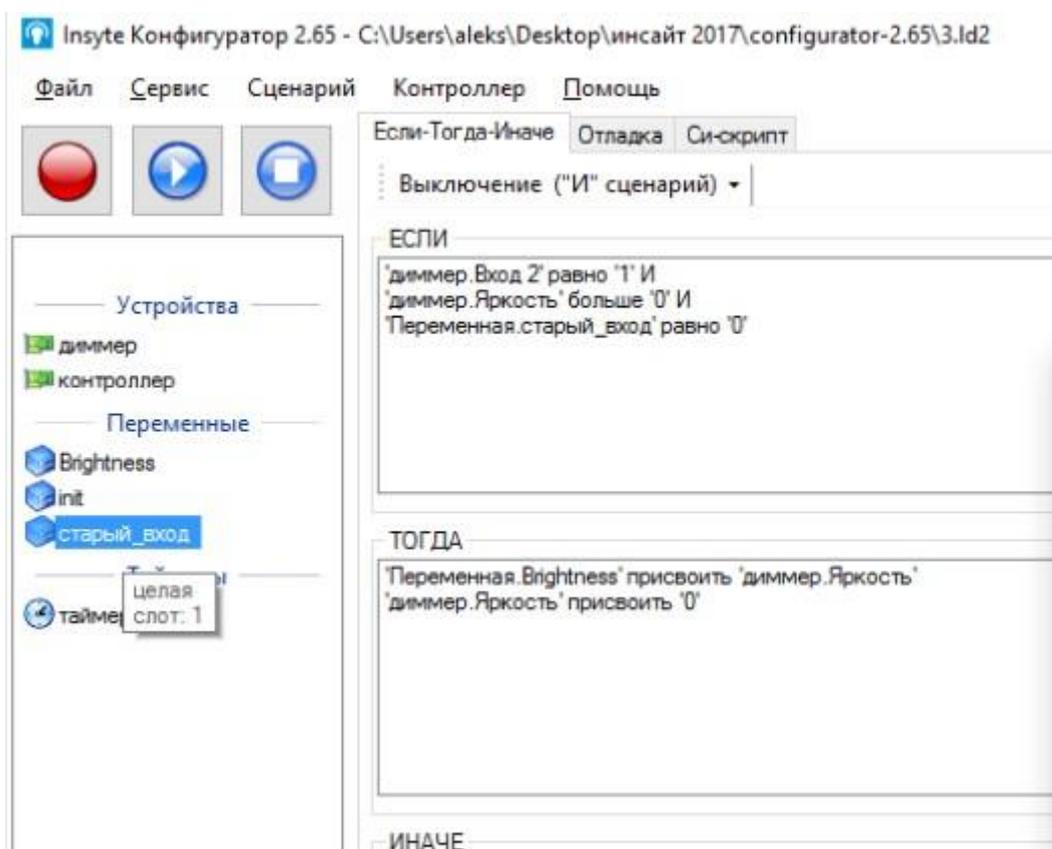


Рисунок 9 – Определение номера слота

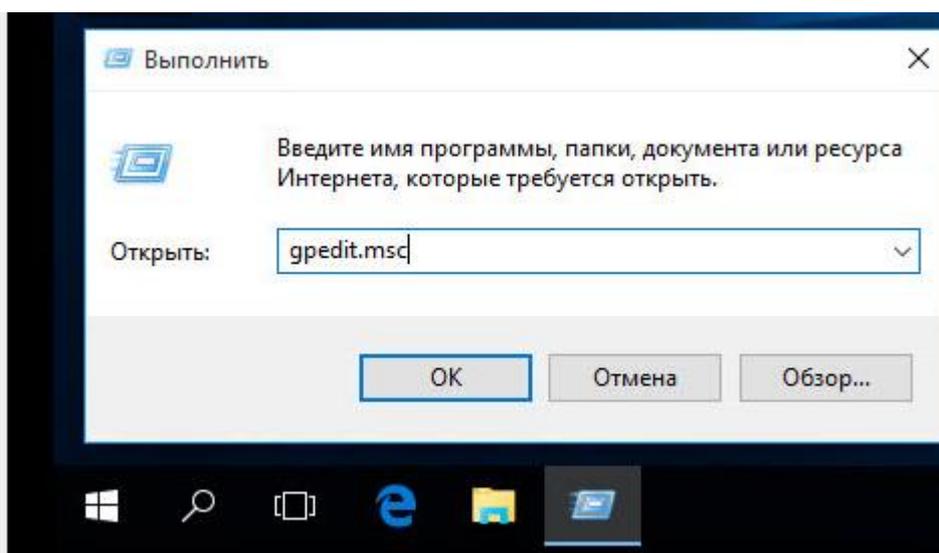
## Подключение контроллера

Перед подключением к компьютеру контроллера необходимо выполнить следующее:

1. Отключить проверку цифровой подписи драйверов в Windows
2. Установить драйвера из папки «usb\_driver» - usb\_driver.inf

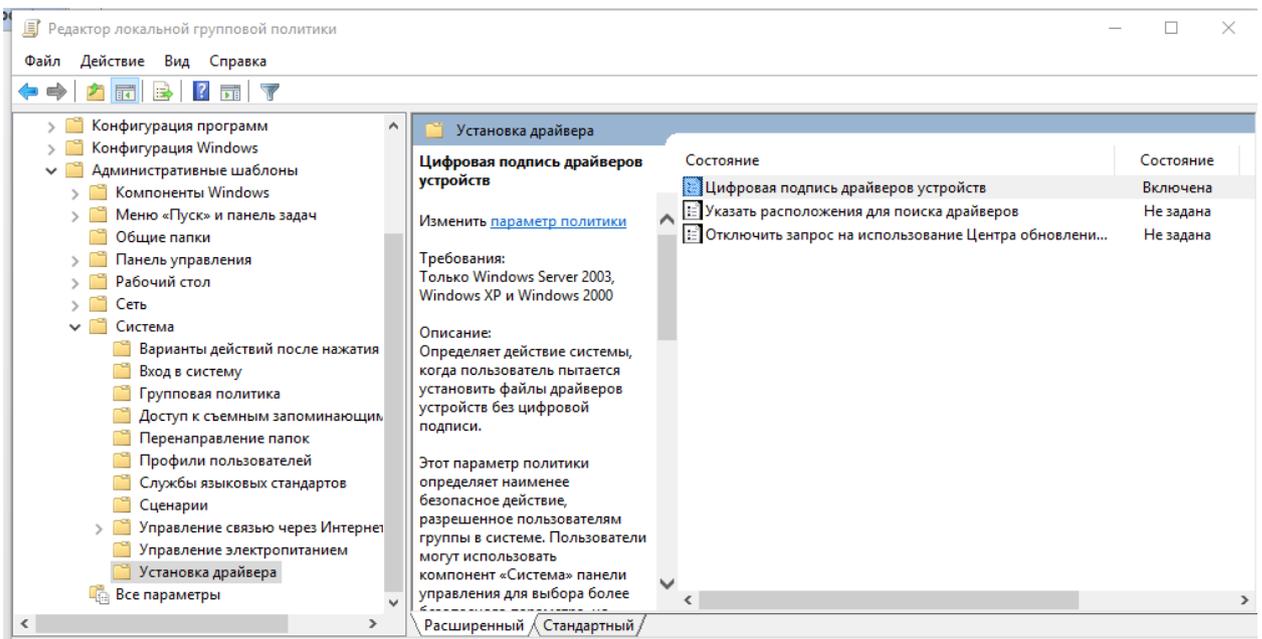
Для этого необходимо выполнить следующий порядок действий:

1. Выполнить запуск редактора локальной групповой политики, нажмите клавиши Win+R на клавиатуре, а затем введите gpedit.msc в окно «Выполнить», нажмите Enter.



У вас откроется редактор групповой политики.

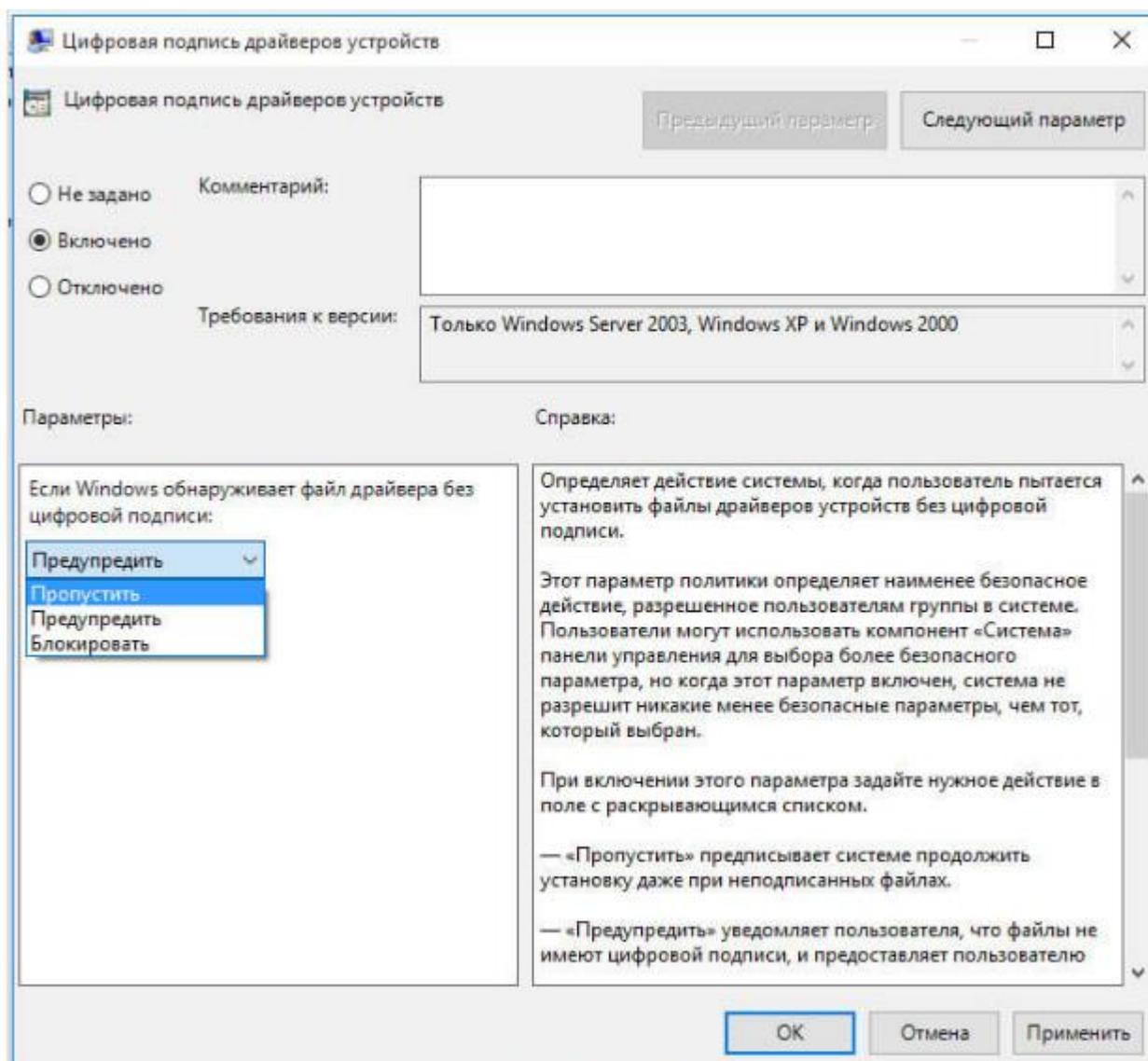
В редакторе перейдите к разделу Конфигурация пользователя - Административные шаблоны - Система- Установка драйвера.



Далее дважды кликните по параметру «Цифровая подпись драйверов устройств» в правой части.

Откроется оно с возможными значениями данного параметра. Убрать их можно двумя способами:

Установить значение «Включено», а затем, в разделе «Если ОС обнаруживает файл драйвера без цифровой подписи» установить «Пропустить».



После установки значений необходимо перезагрузить компьютер.

2. Заходим в диспетчер устройств, в контроллеры USB (например, корневой USB-концентратор) и устанавливаем драйвера из папки «usb\_driver» - usb\_driver.inf.

**INSYTE Electronics** [www.insyte.ru](http://www.insyte.ru)

Телефон службы поддержки покупателей 8-800-25-007-52 (бесплатно по России)