

Rikprom

Руководство
по эксплуатации

Программируемый контроллер

ПРОМЫШЛЕННАЯ ЭЛЕКТРОНИКА



Студия разработки СпецПромДизайн

Разработка электроники и программного обеспечения ...это просто

Web: www.spd.net.ru, E-mail: info@spd.net.ru

СОДЕРЖАНИЕ

ОПИСАНИЕ	3
ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	3
УСТРОЙСТВО И ПРИНЦИП РАБОТЫ	4
ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	7
МЕНЮ НАСТРОЕК	11
ПРОТОКОЛ ОБМЕНА ДАННЫМИ ПО ИНТЕРФЕЙСУ RS-232	17
ПРОТОКОЛ ОБМЕНА WAKE16	21
Описание протокола	21
Структура пакета	21
ПРОГРАММА WakeControl.....	25
ОБНОВЛЕНИЕ ВСТРОЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	29

ОПИСАНИЕ

Программируемый контроллер предназначен для создания систем автоматизированного управления технологическим оборудованием в различных областях промышленности, жилищно-коммунального и сельского хозяйства.

Логику работы контроллера для определённого технологического оборудования можно изменять путём самостоятельной загрузки в него различных версий встроенного программного обеспечения, предоставляемого производителем контроллера.

Контроллер имеет интерфейс RS-232 с гальванической развязкой, через который можно выполнять настройки, управлять выходными сигналами, а также осуществлять обновление встроенного программного обеспечения.

Помимо этого, имеется ещё встроенный порт RS-485 для подключения внешних модулей расширения.

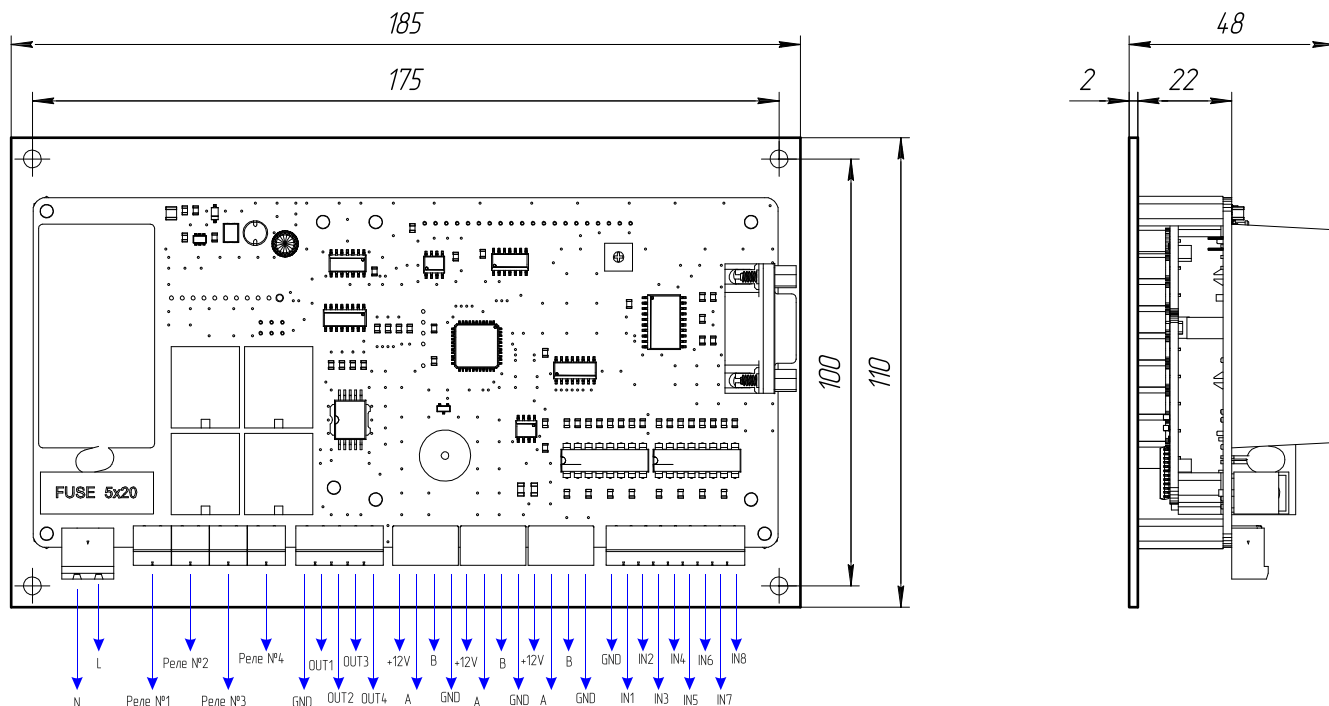


ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

Напряжение питания	~220 В +15% -10%
Максимальная потребляемая мощность	20 Вт
Количество входов	8
Количество релейных выходов.....	4
Количество дискретных цифровых выходов	4
Напряжение цифрового выхода	12 В
Максимальный ток цифрового выхода	1 А
Степень защиты	IP10
Габаритные размеры	185 × 110 × 48 мм
Температурный диапазон работы	от -40°C до +85°C
Относительная влажность воздуха	не более 90% при +35°C

УСТРОЙСТВО И ПРИНЦИП РАБОТЫ

Программируемый контроллер выпускается в панельном варианте с креплением на поверхность электрического шкафа:



Подключение внешних цепей осуществляется при помощи винтовых клеммников. Назначение контактов клеммников следующее:

N, L – питание ~220 В;

Реле №1...Реле №4 – нормально разомкнутые контакты соответствующего электромагнитного реле;

GND – общий провод для дискретных цифровых выходов и входов;

OUT1...OUT4 – дискретные цифровые выходы;

+12В – внутреннее напряжение питания контроллера;

A, B – линии А и В интерфейса RS-485;

IN1...IN8 – входы.



На лицевой панели контроллера расположен графический ЖК-индикатор и 17 светодиодов. Светодиоды с номерами 1...8 в группе «IN» и 1...8 в группе «OUT» отображают состояния соответствующих входов и управляющих линий. При этом номера с 1 по 4 в группе «OUT» соответствуют электромагнитным реле, а номера с 5 по 8 – дискретным цифровым выходам.

Светодиод рядом с символом \triangle индицирует аварийное состояние контроллера. При переводе контроллера в режим загрузчика для обновления встроенного программного обеспечения этот светодиод постоянно мигает с частотой около 2 Гц.

Передняя панель контроллера имеет встроенную плёночную клавиатуру, предназначенную для управления контроллером и настройки ряда параметров через встроенное меню.

Дискретные цифровые линии контроллера позволяют напрямую коммутировать нагрузку с током потребления до 1 А. В активном состоянии они выдают напряжение +12 В.

Все линии имеют защиту от короткого замыкания на «землю». После устранения причины замыкания работоспособность выхода автоматически восстанавливается.

Входы IN1...IN8 предназначены для подключения контактных датчиков с выходом «сухой контакт». Активным сигналом для входов является лог. 0 (замыкание входа на «землю» GND).

Контроллер имеет порт RS-232 с гальванической развязкой, через который осуществляется обмен с компьютером, а также обновление встроенного программного обеспечения.

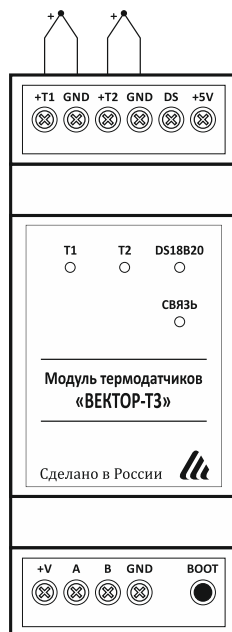
Ко встроенному порту RS-485 можно подключить несколько внешних модулей расширения. В комплект контроллера входят модуль для контроля токов в силовых цепях и модуль термодатчиков.

К модулю контроля токов можно подключить шесть внешних датчиков тока:



Сквозь датчики продеваются фазные провода, питающие нагрузку. В случае если происходит перегорание нагрузки или срабатывание автоматического выключателя, датчик определяет отсутствие тока в цепи и модуль контроля токов выдаёт соответствующих сигнал в контроллер.

Модуль термодатчиков позволяет подключать одновременно две термодпары К-типа:



Он производит их оцифровку с автоматической компенсацией «холодного сая» и передаёт измеренное значение температуры в контроллер.

Показания температуры могут использовать для организации терморегулятора. В этом случае в качестве исполнительного элемента выступает твердотельное реле, управлением которым осуществляется дискретным цифровым выходом контроллера.

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Базовая версия встроенного программного обеспечения предназначена для управления циклической работой станка с тремя исполнительными механизмами:

1. Двигатель подачи детали для обработки.
2. Галогеновая лампа ударного действия для нагрева детали.
3. ТЭН для подогрева воздуха.

Двигатель и галогеновые лампы подключаются через внешние контакторы, рассчитанные на соответствующий ток, а ТЭНы управляются при помощи твердотельного реле, входящего в комплект контроллера.

Факт перегорания лампы или ТЭНа определяется при помощи внешнего блока контроля токов «ВЕКТОР-ДТ6».

Управление нагревом воздуха может осуществляться двумя способами:

1. Регулировка длительностью включения ТЭНов без использования датчика температуры. В этом случае поддержание нужной температуры осуществляется путём задания длительности включённого состояния твердотельного реле и паузы между включениями.
2. При помощи терморелы К-типа, входящей в состав контроллера. При этом реализуется обычный режим терморегулятора с контрольной точкой температуры и гистерезисом.

В процессе работы контроллер использует следующие входы и управляющие выходы:

IN1 – кнопка-грибок аварийного «СТОП». Активным является разомкнутое состояние.

IN2 – концевик начального положения детали.

IN3 – концевик рабочего положения детали.

IN4 – кнопка «ПУСК» для запуска цикла работы.

Реле №1 – двигатель подачи детали на обработку.

Реле №2 – галогеновые лампы ударного действия.

OUT1 – лампа «АВАРИЯ».

OUT4 – твердотельное реле для управления ТЭНами.



При настройке контроллера на работу с одним концевиком его можно подключать к любому входу (IN2 или IN3).

При размыкании входа IN1, к которому подключена кнопка «СТОП», происходит перевод контроллера из любого режима работы в режим «СТОП». В данном режиме все выходы переводятся в неактивное состояние для выключения исполнительных механизмов, а на экране контроллера выводится следующая информация:




Выход из режима «СТОП» осуществляется замыканием входа IN1.



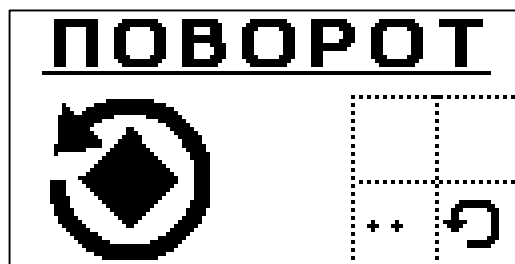
При этом контроллер включает двигатель подачи детали для поворота его в начальное положение!

При замкнутом состоянии входа IN1 контроллер находится в режиме «РАБОТА», в котором происходит ожидание нажатия кнопки «ПУСК» для запуска основного рабочего цикла, а также осуществляется поддержание температуры воздуха. Внешний вид экрана в данном режиме показан ниже:



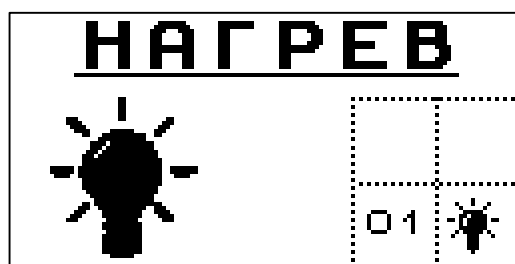
Здесь отображается текущее значение температуры воздуха (при использовании термомпары) и факт включения ТЭНов. Символ  в правом нижнем углу обозначает готовность к пуску.

При нажатии кнопки «ПУСК» осуществляется поворот детали в рабочую зону при помощи двигателя подачи. Поворот осуществляется до срабатывания концевика рабочего положения детали, подключённого ко входу IN3. Во время работы двигателя на экран контроллера выводится следующая информация:

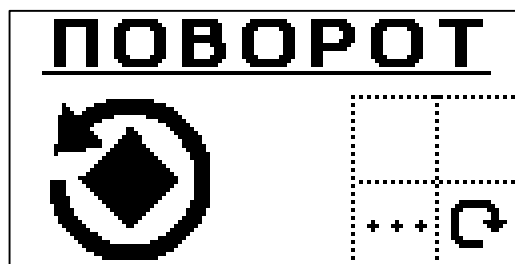


Уменьшающееся количество точек в правом нижнем углу обозначает оставшуюся величину там-аута срабатывания концевика. Если за это время срабатывания не произойдёт, контроллер автоматически обесточит двигатель и переведёт контроллер в режим «АВАРИЯ».

Если срабатывание концевика уложилось в заданное время тайм-аута, то контроллер перейдёт в режим «НАГРЕВ», в котором двигатель останавливается, и при этом включаются галогеновые лампы ударного действия. На экран контроллера в данном режиме выводится следующая информация:



В правом нижнем углу отображается оставшееся время работы галогеновых ламп. По истечению данного времени лампы выключаются, а двигатель начинает дальнейший поворот детали в начальное положение:



При этом контролируется срабатывание концевика начального положения, подключённого ко входу IN2. В случае превышения тайм-аута срабатывания концевика контроллер переходит в режим «АВАРИЯ».

Находясь в любом режиме работы кроме режима «СТОП» контроллер осуществляет проверку подключённых внешних устройств и модулей расширения. При возникновении какой-либо неисправности он переходит в режим «АВАРИЯ», а на экран выводится текстовое описание ошибки и её код:



Если ошибок несколько, то они автоматически отображаются одна за другой через определённое время. После кода ошибки через символ «/» указывается общее число возникших ошибок.

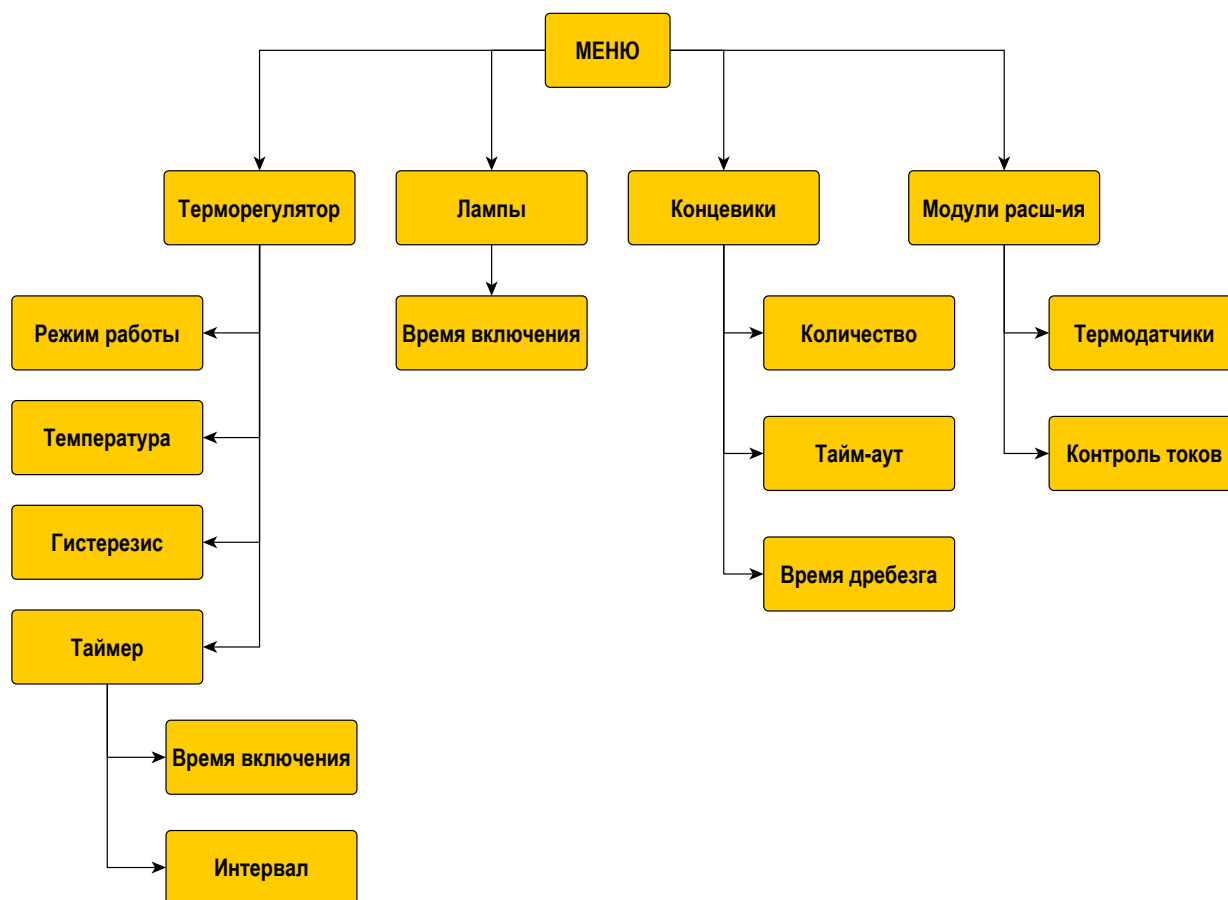
В данном режиме все выходы переводятся в неактивное состояние для выключения исполнительных механизмов.

Выйти из этого режима можно только переходом в режим «СТОП» (размыканием входа IN1). При этом происходит сброс всех ошибок. Но если они возникают снова, контроллер опять переходит в режим «АВАРИЯ».

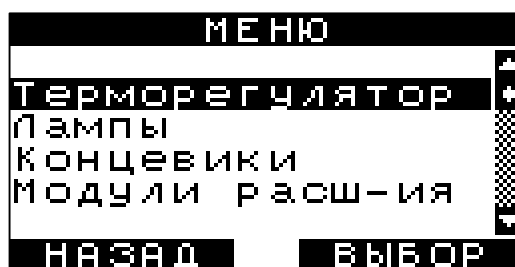
МЕНЮ НАСТРОЕК

Переход в меню осуществляется нажатием кнопки «MENU».

Структура меню показана ниже:



На ЖК-экране контроллера меню отображается в виде списка:

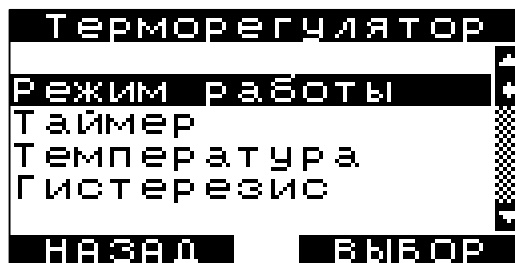


Перемещение курсора на нужный пункт меню осуществляется при помощи кнопок «↑» и «↓». Переход в дочернее меню или переключение в режим редактирования значения параметра производится нажатием кнопки «⇒» или «ENTER», возврат в предыдущее меню – нажатием кнопки «⇐» или «ESC».

В режиме редактирования значение параметра можно вводить либо при помощи цифровых клавиш «0»... «9», либо увеличивать и уменьшать на единицу нажатием кнопок «↑» и «↓».

Далее будут рассмотрены подробно все пункты меню.

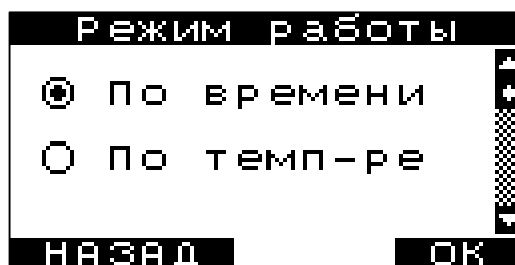
«*Терморегулятор*» – позволяет выбрать режим работы терморегулятора, а также задавать соответствующие параметры.



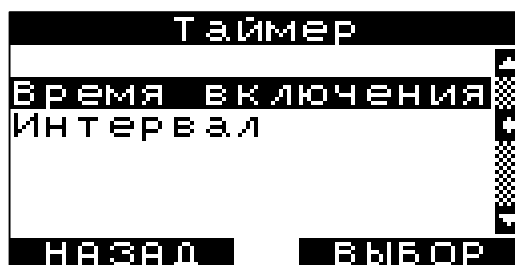
«*Режим работы*» – задаёт режим работы терморегулятора (по времени или по температуре).

В первом случае поддержание температуры определяется длительностью включения ТЭНов без использования датчика температуры. При этом задаётся длительность включённого состояния твердотельного реле и паузы между включениями.

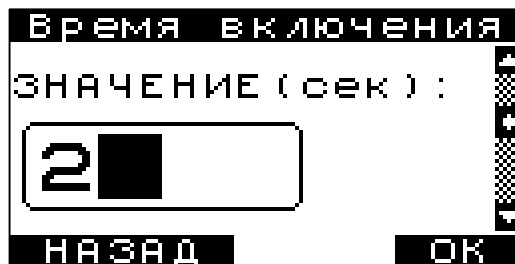
Во втором случае поддержание температуры осуществляется при помощи термомпары К-типа. При этом реализуется обычный режим терморегулятора.



«*Таймер*» – содержит параметры работы терморегулятора в режиме поддержания температуры по времени.



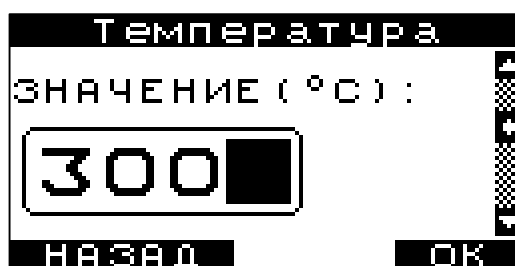
«*Время включения*» – время включённого состояния твердотельного реле.



«*Интервал*» – интервал между включениями твердотельного реле.



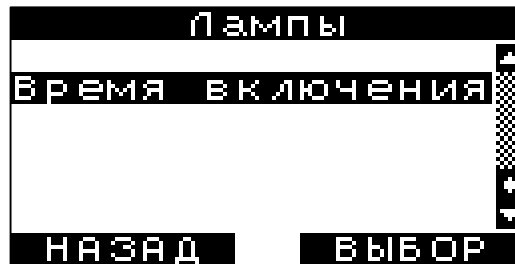
«*Температура*» – контрольная точка температуры в режиме регулировки при помощи термопары.



«*Гистерезис*» – значение гистерезиса в режиме регулировки при помощи термопары.



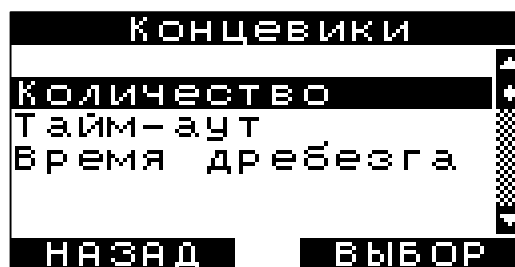
«**Лампы**» – содержит параметры включения ламп ударного действия.



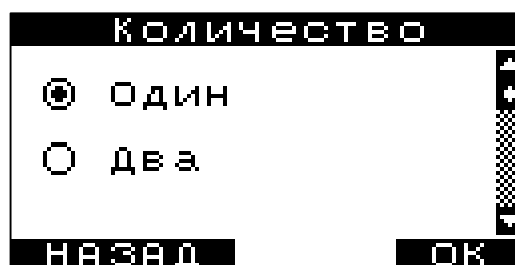
«**Время включения**» – время включённого состояния ламп ударного действия.



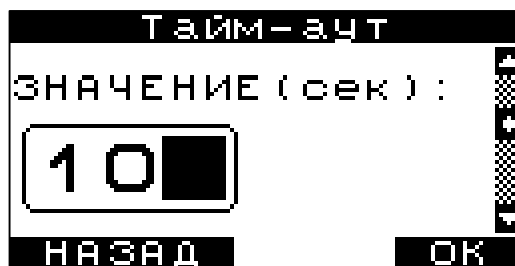
«**Концевики**» – содержит параметры концевиков, по которым осуществляется остановка двигателя подачи детали на обработку.



«**Количество**» – выбор одного концевика (срабатывает как в начальном, так и в рабочем положении) или двух (разные концевики для начального и рабочего положения).

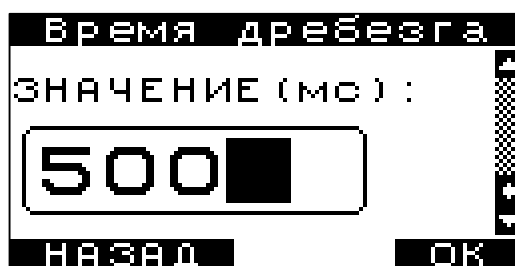


«**Тайм-аут**» – максимальное время, отводимое на срабатывание концевика при повороте двигателя подачи детали. В случае превышения данного времени возникнет соответствующая ошибка.

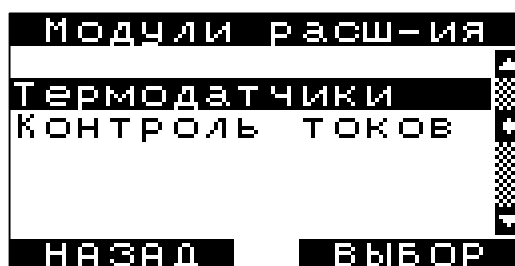


«Время дребезга» – время подавления «дребезга» контактов концевиков, в течение которого контроллер не будет реагировать на их срабатывания. Данный параметр актуален для механических концевиков, срабатывание которых сопровождается многократными кратковременными переключениями контактов, на которые может успеть среагировать контроллер. Время подбирается опытным путём в диапазоне от 200 до 2000 мс.

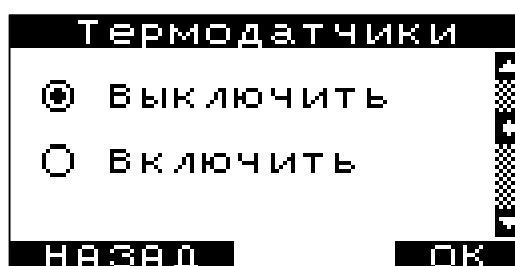
При использовании электронных концевиков, у которых «дребезг» отсутствует, данное время можно задать нулевым.



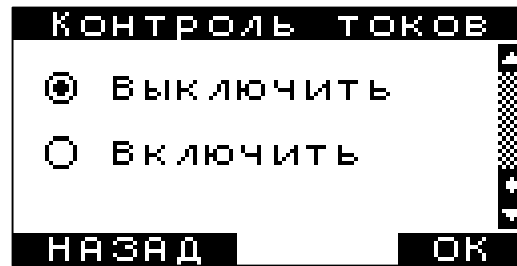
«Модули расширения» – настройка внешних модулей расширения, подключаемых к контроллеру через интерфейс RS-485.



«Термодатчики» – включение/выключение модуля «ВЕКТОР-ТЗ» для работы с термопарами К-типа.



«Контроля токов» – включение/выключение модуля «ВЕКТОР-ДТ6» для работы с внешними датчиками тока.



ПРОТОКОЛ ОБМЕНА ДАННЫМИ ПО ИНТЕРФЕЙСУ RS-232

Настройка контроллера может осуществляться через интерфейс RS-232 по протоколу WAKE16 на скорости 115200 бит/сек. Для управления используются 13 команд. Если контроллер примет пакет с неверным форматом команды, то он передаст ответ с командой 0x22 и пустыми данными.

По умолчанию адрес контроллера имеет значение 0x7FFF (32767). Также может использоваться широковещательный адрес 0x0000 (0). Ответ от контроллера всегда содержит нулевое значение поля адреса, что позволяет всегда отличать его от остальных принимаемых пакетов данных.

Ниже приводится описание команд, реализованных в контроллере. Данные размерностью более одного байта помечаются числом битов в нижнем индексе (например, <номер_блока₁₆>). Многобайтные значения передаются старшими байтами вперёд. Строки передаются в формате языка Си – с завершающим нулём. Символьные и строковые данные передаются в кодировке Windows-1251.

cmReadAddr – чтение адреса контроллера.

Описание: возвращает текущий адрес контроллера.

Команда: 0x0A

Данные: нет

Ответ:

Команда: 0x33

Данные: <addr₁₆> – текущий адрес контроллера

cmReset – сброс контроллера.

Описание: выполняет аппаратный сброс контроллера.

Команда: 0x08

Данные: нет

Ответ:

Команда: 0x33

Данные: нет

cmRunBoot – переключение в режим загрузчика.

Описание: переключает контроллер в режим загрузчика для обновления встроенного программного обеспечения.

Команда: 0x02

Данные: нет

Ответ:

Команда: 0x33

Данные: нет

cmSetOuts – включение/отключение выходных линий.

Описание: задаёт индивидуальное состояние каждой выходной линии.

Команда: 0x51

Данные: <outs_mask >

Значение <outs_mask> содержит маску включения выходных линий. Установленный бит в 0-ом разряде соответствует включённому состоянию реле №1, в 1-ом разряде – реле №2, во 2-ом разряде – реле №1, в 3-ом разряде – реле №4, в 4-ом разряде – активному состоянию линии OUT1, в 5-ом разряде – OUT2, во 6-ом разряде – OUT3, в 7-ом разряде – OUT4.

Ответ:

Команда: 0x33

Данные: нет

cmGetOutsInputs – текущее состояние входов и выходных линий.

Описание: считывает текущее состояние входов и выходных линий.

Команда: 0x62

Данные: нет

Ответ:

Команда: 0x33

Данные: <outs_mask>, <inputs_mask>

Значение <outs_mask> содержит маску выходных линий. Установленный бит в 0-ом разряде соответствует включённому состоянию реле №1 в 1-ом разряде – реле №2, в 2-ом разряде – реле №3, в 3-ом разряде – реле №4, в 4-ом разряде – активному состоянию линии OUT1, в 5-ом разряде – OUT2, в 6-ом разряде – OUT3, в 7-ом разряде – OUT4

Значение <inputs_mask> содержит маску входов. Установленный бит в 0-ом разряде соответствует активному состоянию входа IN1, в 1-ом разряде – IN2, во 2-ом разряде – IN3 и т.д.

cmGetInfo – получение информации об устройстве.

Описание: считывает информацию о названии устройства, количестве входов, выходов и т.п..

Команда: 0x71

Данные: нет

Ответ:

Команда: 0x33

Данные:

<mode> – код режима работы (0x11 – основной режим, 0x10 – режим загрузчика)

<version> – номер версии программного обеспечения

<build₁₆> – номер сборки программного обеспечения

<str₀>, <str₁>...<0x00> – текстовая строка с названием устройства

0x0C – сигнатура микроконтроллера (PIC18F46K22)

0x01, 0x00, 0x00 – объем внешней памяти (64 кБ)

0x04 – сигнатура блока данных (целое число)

<str₀>, <str₁> ... <0x00> – название блока «Режим работы»

0x02 – размер данных

<mode₁₆> – 1 – «РАБОТА», 2 – «ПОВОРОТ», 3 – «СТОП», 4 – «НАГРЕВ»,
5 – «ВОЗВРАТ», 6 – «АВАРИЯ»

0x04 – сигнатура блока данных (целое число)

<str₀>, <str₁> ... <0x00> – название блока «Кол-во выходов»

0x02 – размер данных

<outs_cnt₁₆> – количество выходов (8)

0x04 – сигнатура блока данных (целое число)

<str₀>, <str₁> ... <0x00> – название блока «Кол-во входов»

0x02 – размер данных

<inputs_cnt₁₆> – количество входов (8)

0x04 – сигнатура блока данных (целое число)

<str₀>, <str₁> ... <0x00> – название блока «Сост-е выходов»

0x02 – размер данных

<outs_state₁₆> – текущее состояние выходов

0x04 – сигнатура блока данных (целое число)

<str₀>, <str₁> ... <0x00> – название блока «Сост-е входов»

0x02 – размер данных

<inputs_state₁₆> – текущее состояние входов

0x05 – сигнатура блока данных (длинное целое число)

<str₀>, <str₁> ... <0x00> – название блока «Счётчик наработки»

0x04 – размер данных

<ticks₃₂> – значение счётчика в секундах

0x01 – сигнатура блока данных (строка)

<str₀>, <str₁> ... <0x00> – название блока «DateTime FW»

<size> – размер данных (длина строки + 1)

<str₀> ... <str_{size-1}>, <0x00> – текстовая строка даты и времени

компиляции встроенного программного обеспечения

микроконтроллера

ПРОТОКОЛ ОБМЕНА WAKE16

Описание протокола

Протокол WAKE16 является логическим уровнем интерфейса управления оборудованием с помощью асинхронного последовательного канала. Физический уровень интерфейса протоколом не определяется. В качестве него может использоваться, например, RS-232 или RS-485. Протокол позволяет производить обмен пакетами данных длиной до 2^{16} байт с адресуемыми устройствами, которых может быть до $2^{15} - 1$. Последовательный канал должен быть сконфигурирован следующим образом:

- 1) число бит в посылке – 8;
- 2) количество стоп-бит – 1;
- 3) бит чётности – нет;
- 4) скорость обмена – 1200...921600 бит/сек;
- 5) использование линий управления модемом – произвольное.

Основой протокола WAKE16 является протокол SLIP (UNIX™ Serial Link Interface Protocol). Передача данных осуществляется в двоичном виде, то есть используются все возможные значения байта (0x00...0xFF). Для передачи служебной информации зарезервированы два кода: FEND = 0xC0 (Frame End) и FESC = 0xDB (Frame Escape). Управляющий код FEND служит для обозначения начала посылки, а код FESC служит для передачи ESC-последовательностей. Если в потоке данных встречаются байты, значения которых совпадают с управляющими кодами, производится подмена этих байт ESC-последовательностями. Этот механизм называют байт-стаффингом. Код FEND заменяется последовательностью <FESC>, <TFEND>, а код FESC – последовательностью <FESC>, <TFESC>, где TFEND = 0xDC (Transposed FEND), TFESC = 0xDD (Transposed FESC). Коды TFEND и TFESC являются управляющими только в ESC-последовательностях, поэтому при передаче данных они в подмене не нуждаются.

Структура пакета

Пакет всегда начинается управляющим кодом FEND (0xC0). Затем следует необязательный байт адреса, после которого идёт байт команды. За ним следует байт количества данных и собственно байты данных. Завершает пакет байт контрольной суммы CRC16:

FEND	ADDR	CMD	N	Data ₁	...	Data _N	CRC16
------	------	-----	---	-------------------	-----	-------------------	-------

FEND: управляющий код FEND (C0h) является признаком начала пакета. Благодаря стаффингу, этот код больше нигде в потоке данных не встречается, что позволяет в любой ситуации однозначно определять начало пакета.

ADDR: 16-битное слово адреса используется для адресации отдельных устройств. На

практике распространена ситуация, когда управление осуществляется только одним устройством. В таком случае слово адреса не требуется, и его можно не передавать. Вместо него сразу за кодом FEND передаётся байт команды CMD.

Часто встречается ситуация, когда в сети имеется всего одно устройство, которое может инициировать обмен данными (мастер-устройство), и при этом обмен ведётся по принципу «запрос – ответ». Примером могут являться сети, построенные на основе интерфейса RS-485. В этом случае пакет данных от мастера содержит адрес устройства, которому этот пакет предназначен. Однако в ответном пакете указывать адрес мастера нет необходимости, так как он единственный в сети.

Для того чтобы можно было однозначно установить, к адресу или команде относится второй байт пакета, введены некоторые ограничения. Для адресации используется 15 бит, а старший бит, передаваемый вместе с адресом, должен всегда быть установлен в единицу:

	D15	...	D1	D0
ADDR =	1	...	A1	A0

Иногда возникает необходимость передать какую-то команду или данные сразу всем устройствам. Для этого предусмотрен коллективный вызов, который осуществляется путём передачи нулевого адреса (учитывая единичный старший бит, в этом случае передаваемый адрес равен 0x8000). Нужно отметить, что передача в пакете нулевого адреса полностью аналогична передаче пакета без адреса. Поэтому при реализации протокола можно автоматически исключать нулевой адрес из пакета. Учитывая разрядность адреса и один зарезервированный адрес для коллективного вызова, максимальное количество адресуемых устройств составляет $2^{15} - 1$.

Если возникает необходимость передать значение адреса, содержащие значения 0x40 или 0x5B (передаваемый байт в этом случае будет равен 0xC0h или 0xDB), то производится байт-стаффинг. Поэтому следует учитывать, что устройства с такими адресами требуют большей длины пакета. Это может быть заметно в тех случаях, когда используются короткие пакеты. В таких случаях следует избегать назначения устройствам названных адресов.

CMD: байт команды всегда должен иметь нулевой старший бит:

	D7	D6	D5	D4	D3	D2	D1	D0
CMD =	0	C6	C5	C4	C3	C2	C1	C0

Таким образом, код команды занимает 7 бит, что позволяет передавать до 128 различных команд. Коды команд выбираются произвольно в зависимости от нужд приложения. Рекомендуется использовать несколько стандартных кодов команд:

Код	Описание команды
0x 01	Передача управления основной программе центрального контроллера
0x 02	Передача управления загрузчику центрального контроллера
0x 03	Чтение блока FLASH-памяти центрального контроллера
0x 04	Запись блока FLASH-памяти центрального контроллера
0x 08	Сброс устройства
0x 71	Запрос информации об устройстве
0x7A	Изменение адреса устройства

Команды обычно имеют несколько параметров, которые передаются далее в виде пакета данных.

Поскольку код команды всегда имеет нулевой старший бит, этот код никогда не совпадает с управляющими кодами. Поэтому при передаче команды байт-стаффинг никогда не производится.

N: имеет значение, равное числу передаваемых байт данных. Под величину N отводится два байта:

	D15	...	D1	D0
N =	0	...	N1	N0

Таким образом, один пакет может содержать до 2^{16} байт данных. Значение N не учитывает служебные байты пакета FEND, ADDR, CMD, N и CRC16. В результате байт-стаффинга фактическая длина пакета может возрасти. Значение N не учитывает этот факт и отражает количество полезных байт данных (то есть значение N всегда таково, как будто байт-стаффинг не осуществляется). Если передаваемая команда не имеет параметров, то передаётся $N = 0x0000$ и байты данных опускаются.

Если возникает необходимость передать значение N, равное $0xC0$ или $0xDB$, то производится байт-стаффинг. Однако при таких больших значениях N длина пакета столь велика, что его удлинение ещё на один байт практически незаметно.

Data₁...Data_N: байты данных, количество которых определяется значением N. При $N = 0$ байты данных отсутствуют. Байты данных могут иметь любое значение, кроме FEND ($0xC0$) и FESC ($0xDB$). Если возникает необходимость передать одно из этих значений, то производится байт-стаффинг.

CRC: 16-битное слово контрольной суммы CRC16. Контрольная сумма рассчитывается перед операцией байт-стаффинга для всего пакета, начиная с байтов адреса (или байта команды, если адреса нет) и заканчивая последним байтом данных. При вычислениях используется значение адреса вместе со старшим битом.

Для расчёта контрольной суммы используется полином $CRC16 = X^{16} + X^{12} + X^5 + 1$

([1] 0001 0000 0010 0001 = 0x1021). Значение CRC16 перед вычислением инициализируется числом 0xFFFF. Если CRC16 содержит значения байтов 0xC0 и 0xDB, то они заменяются ESC-последовательностями.

Функция для вычисления CRC16 показана ниже (используется «зеркальный» полином 1000 0100 0000 1000 [1] = 0x8408):

```
void UpdateCRC16 (char b)
{
    char i;

    CRC16 ^= b;

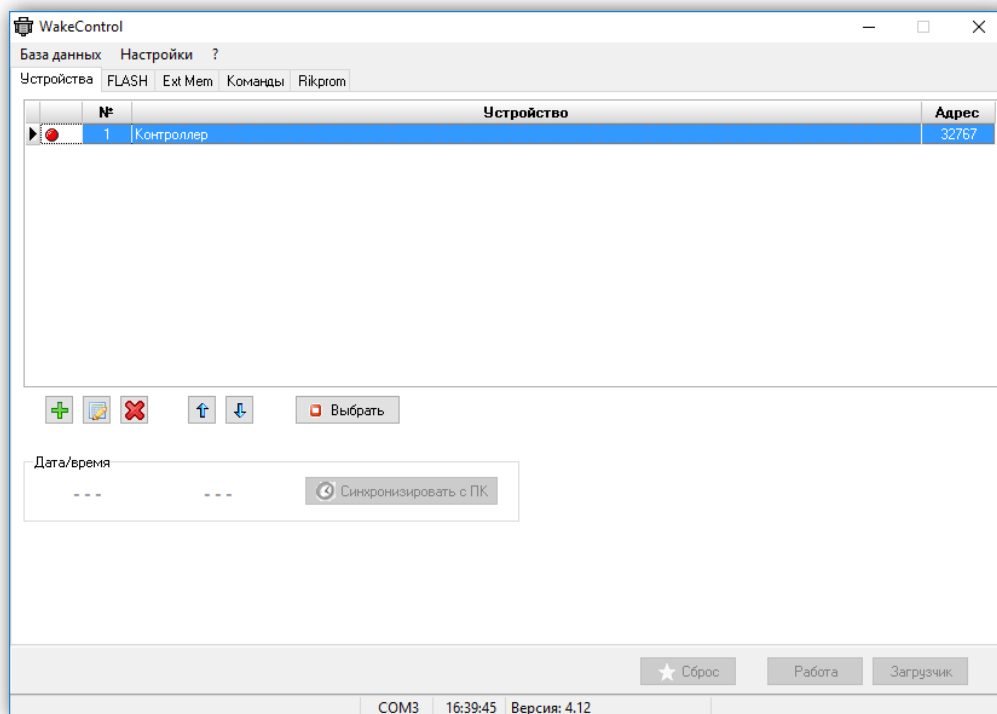
    for (i = 8; i > 0; i--)
        if (CRC16 & 1)
            CRC16 = (CRC16 >> 1) ^ 0x8408;
        else
            CRC16 >>= 1;
}
```

Эту функцию следует вызвать для каждого байта, в результате чего переменная CRC16 будет содержать вычисленное значение контрольной суммы.

ПРОГРАММА WakeControl

Программа WakeControl предназначена для управления контроллером по интерфейсу RS-232, изменения адреса устройства и обновления встроенного программного обеспечения.

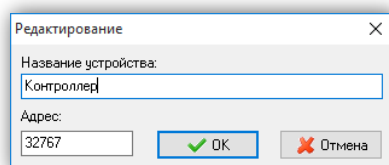
Внешний вид главного окна программы показан ниже:



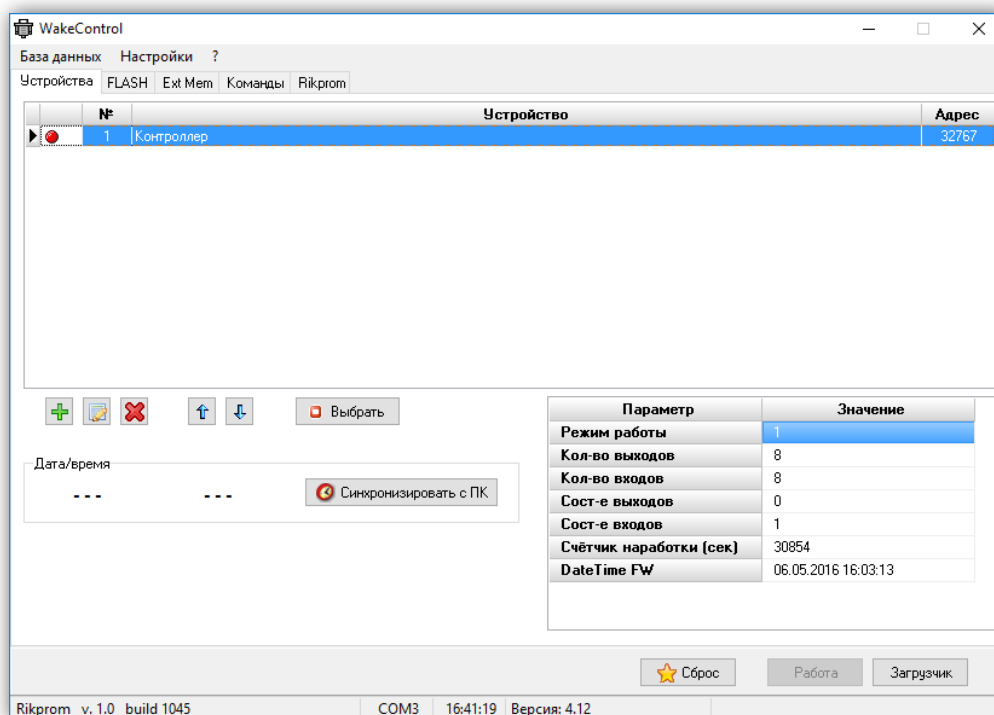
На вкладке «Устройства» отображается список зарегистрированных контроллеров и их адреса. При помощи соответствующих кнопок можно добавлять, изменять и удалять устройства.

Каждый контроллер имеет собственный адрес, который можно сменить через программу WakeControl. По умолчанию адрес всех контроллеров 0x7FFF (32767).

Новое устройство необходимо зарегистрировать. Для этого следует нажать кнопку «Добавить» и в появившемся окне задать название устройства и его адрес:



Чтобы установить связь необходимо выбрать нужное устройство из списка и нажать кнопку «Выбрать». При успешном установлении связи в строке статуса программы появится название устройства (Rikprom) и версия его встроенного ПО:

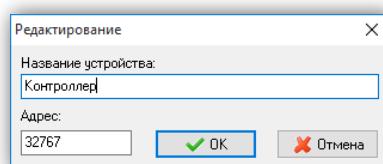


Если вместо названия устройства в строке статуса будет написано «Bootloader», то это означает, что устройство находится в режиме загрузчика. В зависимости от режима работы устройства (рабочий или загрузчик) будут доступны те или иные функции программы WakeControl.

В режиме загрузчика можно только считывать и записывать содержимое FLASH-памяти устройства (для обновления встроенного ПО), а также изменять адрес. В рабочем режиме можно управлять электромагнитными реле и считывать состояние входов.

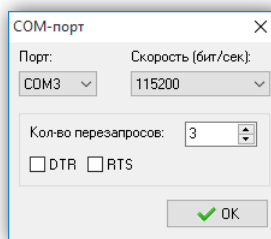
Контроллер можно перевести из рабочего режима в режим загрузчика и обратно при помощи кнопок «Загрузчик» и «Работа». Также можно выполнить общий сброс устройства нажатием кнопки «Сброс».

Чтобы изменить адрес устройства необходимо перевести его в режим загрузчика, а затем в меню «Настройки» выбрать пункт «Сменить адрес...». В появившемся окне необходимо ввести новый адрес и нажать кнопку «ОК»:



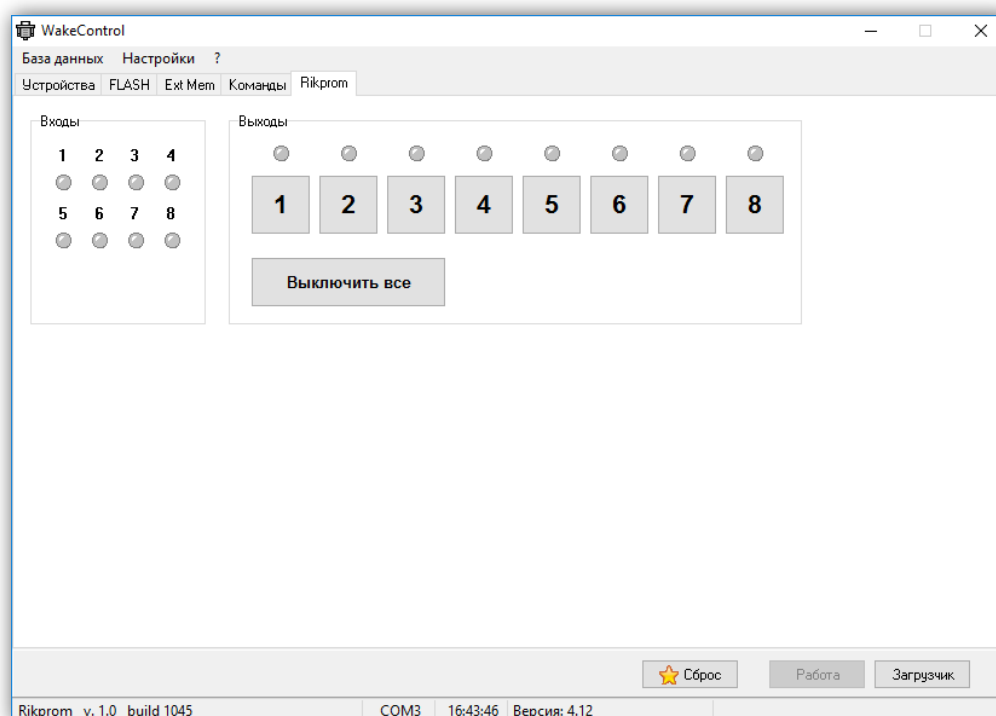
Устройство будет отвечать по новому адресу только после выполнения общего сброса нажатием кнопки «Сброс» или отключением и повторной подачей питания.

Если после запуска программы WakeControl и выбора соответствующего устройства связь не установилась, то необходимо выбрать пункт «COM-порт...» меню «Настройка»:

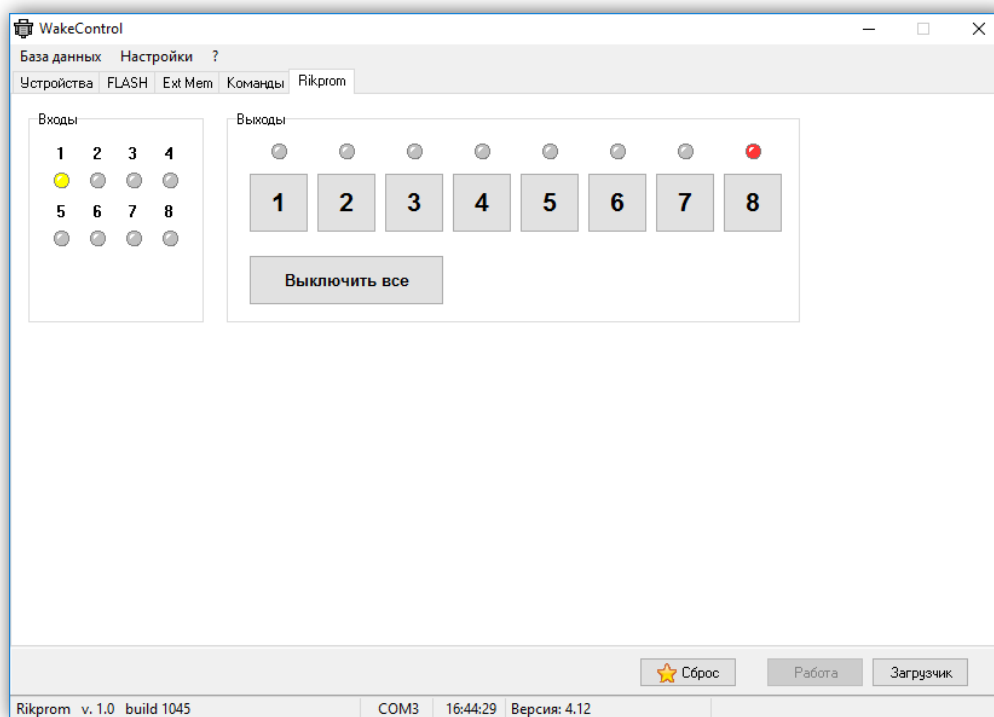


В данном окне следует указать номер СОМ-порта, к которому подключено устройство, а скорость передачи данных задать равной 115200.

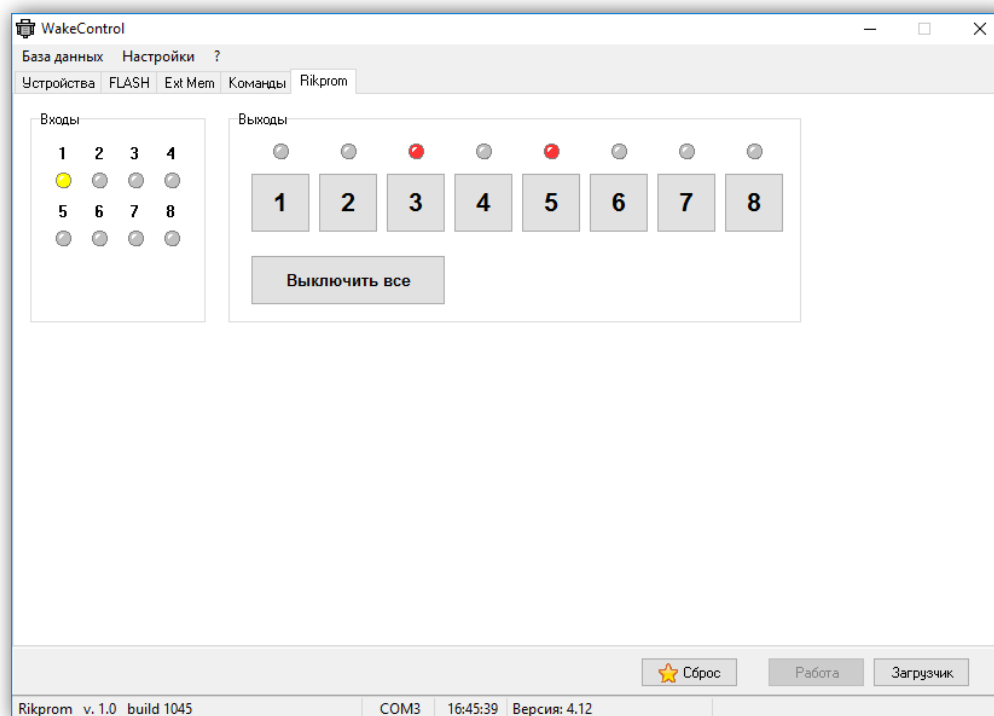
После настройки связи с устройством можно перейти на вкладку «Rikprom» и проверить работоспособность устройства:



Программа WakeControl постоянно опрашивает состояние входов и отображает их состояние в поле «Входы». Сработавшему датчику будет соответствовать жёлтый цвет индикатора:



Управление электромагнитными реле осуществляется нажатием на соответствующую кнопку (1 2 3 4 5 6 7 8). Первое нажатие включит реле, второе – выключит. Включённому состоянию соответствует красный цвет индикатора, расположенного над кнопкой:

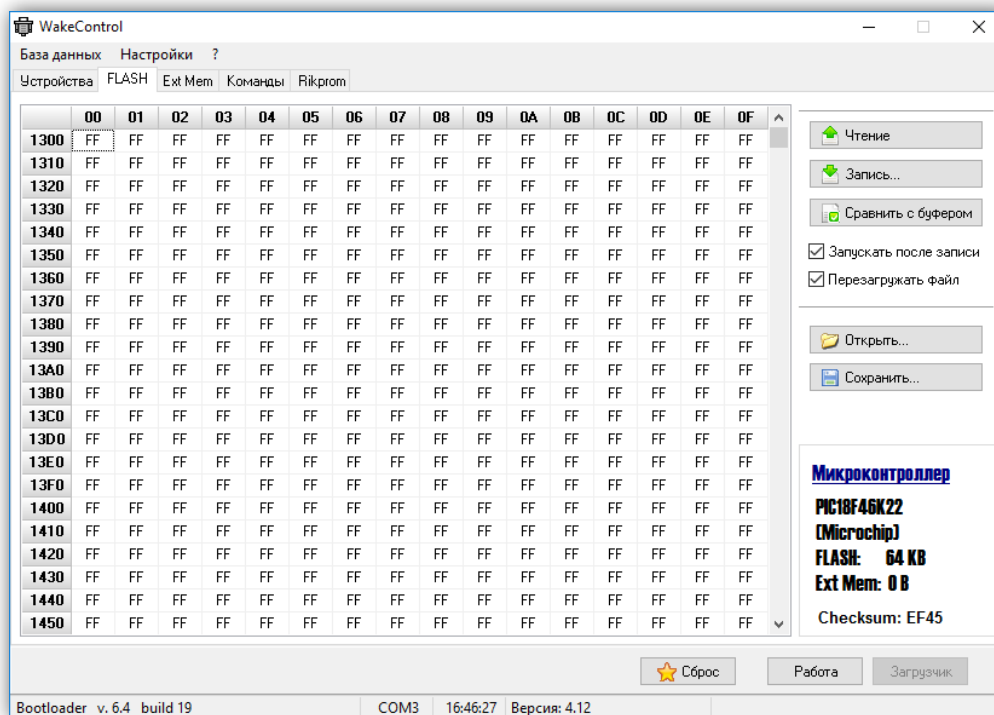


Нажатие на кнопку «Выключить все» приводит к выключению всех электромагнитных реле.

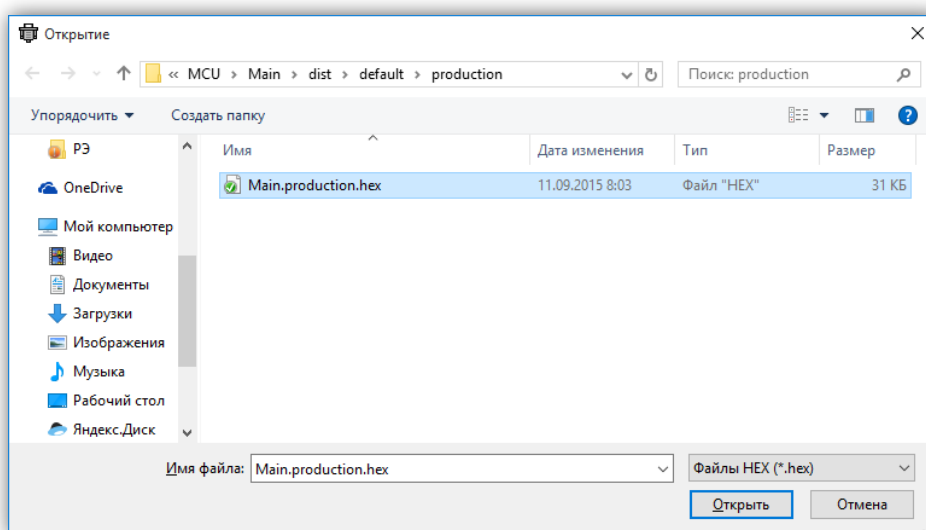
ОБНОВЛЕНИЕ ВСТРОЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Для загрузки нового программного обеспечения в контроллер необходимо выполнить следующие действия:

- 1) Запустить программу WakeControl и перевести устройство в режим загрузчика.
- 2) Перейти на вкладку FLASH и нажать кнопку «Открыть...»:

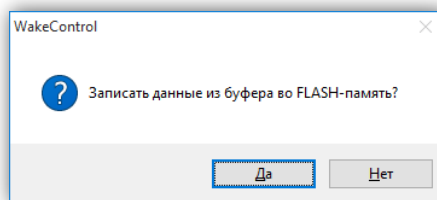


В появившемся окне нужно выбрать соответствующий файл формата Intel HEX:

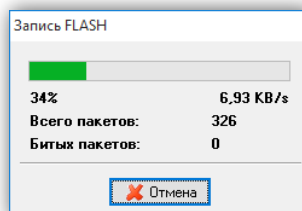


После выбора файла и нажатия кнопки «Открыть» файл будет загружен, а его содержимое в шестнадцатеричном виде будет отображено на панели.

Для записи нового программного обеспечения в устройство необходимо нажать кнопку «Запись...», а затем кнопку «Да» в появившемся диалоговом окне:



При этом начнётся процесс записи:

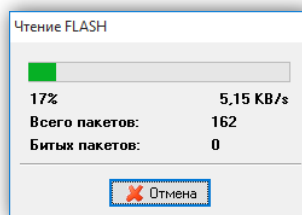


После окончания записи следует нажать кнопку «Работа» для перевода устройства в рабочий режим. Если всё прошло нормально, то в строке статуса программы будет выведена строка с новой версией программного обеспечения.

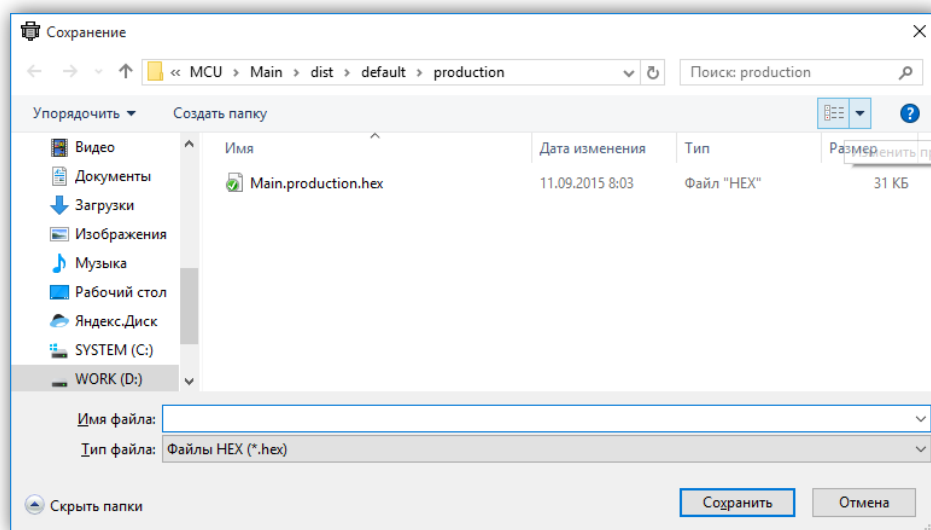


Перед обновлением встроенного программного обеспечения рекомендуется сохранить предыдущую версию!

Действия по сохранению текущей версии встроенного программного обеспечения аналогичны действиям при записи, но после перевода устройства в режим загрузчика следует нажать кнопку «Чтение...». При этом начнётся процесс считывания данных из устройства:



После окончания чтения нужно нажать кнопку «Сохранить...» и в появившемся окне задать имя файла:



Далее необходимо нажать кнопку «Сохранить», после чего текущая версия ПО будет сохранена в заданном файле.