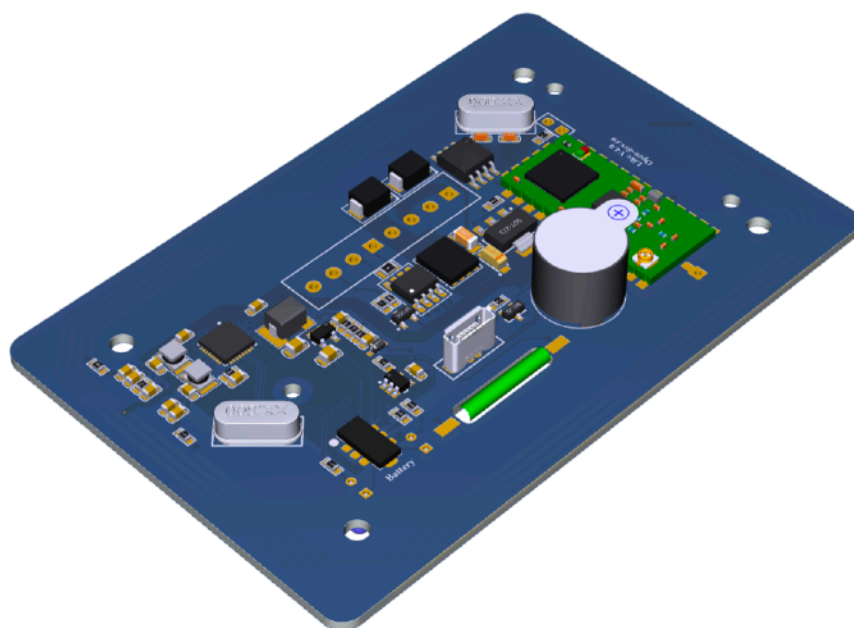


Программируемый RFID контроллер с беспроводным интерфейсом



Руководство пользователя

www.open-dev.ru

main@open-dev.ru

Open-Development

ООО «Открытые разработки»

ПРОГРАММИРУЕМЫЙ RFID КОНТРОЛЛЕР С БЕСПРОВОДНЫМ ИНТЕРФЕЙСОМ

Описание

Считыватель бесконтактных карт предназначен для чтения и записи данных с бесконтактных меток с рабочей частотой 13,56МГц стандарта MIFARE Ultralight, Classic 1K, Classic 4K и передачи данных по беспроводному интерфейсу.

Считыватель позволяет изменять встроенную микропрограмму для получения оптимального режима работы.

Данное устройство представляет собой контроллер с исполняемой программой на языке Python3 в реализации MicroPython[1].

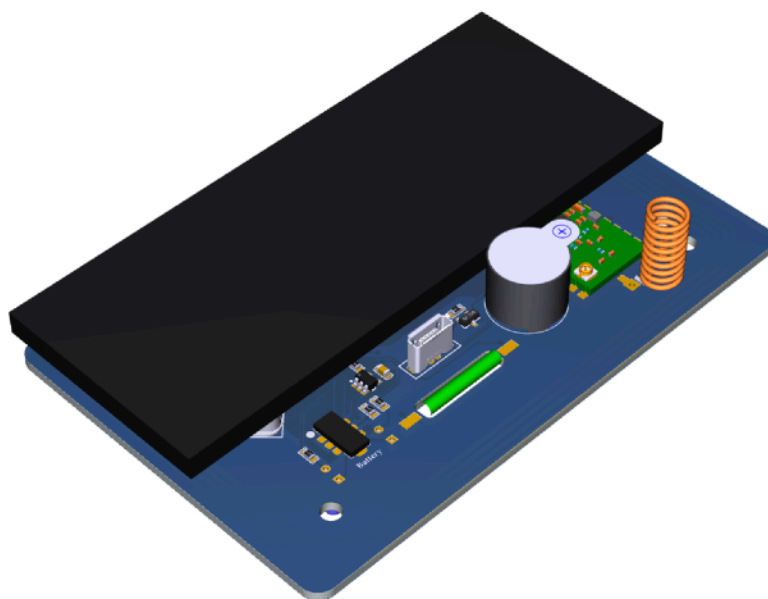


Рис1. Внешний вид устройства с установленным аккумулятором.
Цвет устройства может отличаться.

Основные возможности

- Интерфейсы подключения: RF868, USB.
- Поддерживаемые типы карт: MIFARE Ultralight, Classic 1K, Classic 4K.
- Передача UID и/или заданного блока данных.
- Возможность запуска пользовательской управляющей программы.
- Двусторонняя связь по беспроводному интерфейсу.
- Габариты: 95x62x20 мм .
- Работает в температурном диапазоне -10/+40.
- Светодиодная и звуковая индикация взаимодействия с картой (настраиваемая).
- Питание от USB или от аккумулятора.
- Возможность заряда аккумулятора через usb-интерфейс.
- Контроль заряда и состояния аккумулятора.

Области применения

- Дисконтные и платежные системы;
- Системы контроля доступа;
- Системы идентификации и персонализации;
- Системы автоматизации;
- Пассажирский транспорт;
- Социальные и банковские карты;
- Системы условного доступа в компьютерных сетях;
- Логистика.

Основные узлы.

- 1 - Разъём подключения аккумулятора (опционально).
- 2 - Геркон.
- 3 - Разъём microUSB.
- 4 - Клеммная колодка (опционально).
- 5 - Процессорный модуль.
- 6 - Клемма сброса программного обеспечения.

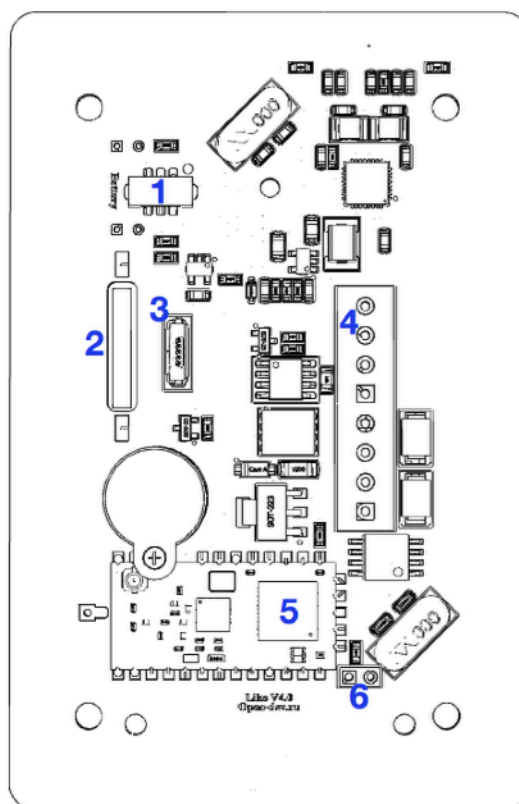


Рис2. Расположение основных узлов устройства

Начало работы.

1. Подключить к считывателю питание по USB или заряженный аккумулятор.
2. Включить устройство поднеся магнит к геркону (Рис2, элемент 2).

Поведение, зависящее от предустановленной программы!

3. Устройство должно произвести двойной светозвуковой сигнал. Это означает, что устройство готово к работе.
4. Поднести к устройству тестовую карту.

5. Успешное чтение карты подтверждается свечением зелёного светодиода, а успешная передача - звуковым сообщением (свечением красного светодиода при отключенной звуковой индикации).
6. RFID считыватель передает посылку вида: <UID><BAT>, где UID - [4..10] байт идентификаторы карты в HEX в текстовом представлении, BAT - уровень заряда аккумулятора в % от полного в HEX (FF - в случае ошибки считывания).
7. Реализованный встроенный фильтр считываемых карт обеспечивает защиту от дубликатов при передаче. Если карта была поднесена повторно до разрешённого времени, загорается красный светодиод, но передачи не происходит. Время работы фильтра указывается в файле настроек.
8. Для приёма беспроводных данных с устройства требуется беспроводной приёмопередатчик с USB интерфейсом, совместимый с данным устройством.

Изменение настроек и управляющей программы.

При подключении к ПК по интерфейсу USB устройство определяется в ОС, как flash-память с меткой ODFLASHxxxx. На встроенной памяти хранится исполняемая программа устройства и дополнительные данные.




Программа состоит из двух обязательных файлов и возможных дополнительных данных.

boot.py - файл, который всегда загружается первым.

main.py - основной файл программы.

settings.cfg - файл настроек для программы.

pybcdd.inf - драйвер последовательного порта (для ОС Windows).

| Имя |
|--|
|  boot.py |
|  pybcdd.inf |
|  main.py |
|  settings.cfg |
|  rfid.py |

По-умолчанию на встроенной карте стоит «защита от записи». При необходимости внести изменения в настройки или управляющую программу требуется предварительно снять защиту от записи. Для этого требуется выполнить следующие действия:

1. Выполнить раздел «Включение устройства без запуска программы», либо подключиться к устройству с помощью программы-терминала по последовательному порту и остановить выполнение программы, нажатием Ctrl-C. После нажатия должно проявиться приглашение в интерактивную консоль «>>>».
2. Ввести команду **pyb.Flash().lock(False)** и нажать «ввод».
3. При правильном исполнении, память, доступная в ОС в виде подключаемого диска, станет доступна к изменениям. Обрато установить защиту можно командой: **pyb.Flash().lock(True)** или перезагрузить устройство.
4. Во избежание порчи данных перед отключением устройства необходимо производить «безопасное извлечение» средствами операционной системы.

Пример файла настроек settings.cfg. Настройки хранятся в формате json.

| Параметр | Обозначение | Значение |
|--|---------------------|----------|
| Адрес устройства (1..254) | SRC_ADDR | 41 |
| Адрес получателя (1..254) | DST_ADDR | 1 |
| Пауза между опросами карты, мсек | READ_TIMEOUT_MS | 1000 |
| Звуковая индикация (0-выкл., 1 - вкл.) | USE_BUZZER | 1 |
| Передача состояния заряда аккумулятора (0-выкл., 1 - вкл.) | USE_BAT_CHECK | 1 |
| Режим пониженного энергопотребления (0-выкл., 1 - вкл.) | USE_SLEEP | 0 |
| Время работы фильтра от повторного поднесения карты, сек | CARD_FILTER_TIMEOUT | 15 |
| Значение UID карты для выключения устройства в hex формате | CARD_OFF_UID | «» |

Пример файла settings.cfg с настройками:

```
{
  "SRC_ADDR": 41,
  "DST_ADDR": 1,
```

```
"READ_TIMEOUT_MS": 1000,  
"USE_BUZZER": 1,  
"USE_BAT_CHECK": 1,  
"CARD_FILTER_TIMEOUT": 15,  
"USE_SLEEP": 0,  
"CARD_OFF_UID": "CCF9E907"  
}
```

Дополнительные сведения о программировании устройства можно подчерпнуть из **руководства по языку**[2].

При отключенной защите от записи: при изменении содержимого встроенной карты памяти требуется всегда выполнять «безопасное извлечение» при отключении устройства от ПК. Небезопасное извлечение может привести к повреждению встроенной программы.

Включение устройства без запуска программы.

Для включения устройства без исполнения программы, записанной в main.py, необходимо:

1. Замкнуть контакты разъёма 6.
2. Включить устройство.
3. Убедиться, что красный светодиод горит, а зелёный мигает.
4. В момент, когда зелёный светодиод потушен, разъединить контакты разъёма 6.

Сброс внутренней flash-памяти в исходное состояние.

Для восстановления внутренней flash-памяти необходимо:

1. Замкнуть контакты разъёма 6.
2. Включить устройство.
3. Убедиться, что красный светодиод горит, а зелёный мигает.
4. В момент, когда зелёный светодиод горит, разъединить контакты разъёма 6.

Подзарядка встроенного аккумулятора.

При питании устройства от USB происходит автоматическая зарядка подключенного аккумулятора. Среднее время полного заряда аккумулятора током 400мА составляет 4-6 часов.

Технические характеристики.

| Параметр | Значение |
|----------------------------|---|
| Напряжение питания | DC 5В |
| Интерфейс | USB |
| Метки стандарта: | MIFARE Ultralight, Classic 1K, Classic 4K |
| Дальность считывания меток | до 6 см |
| Температура хранения | -10 .. +40 C |
| Габаритные размеры, мм | 95x62x20 |

Гарантия.

Гарантийный срок 6 месяцев с продажи конечному потребителю.

Неисправности, выявленные в течение гарантийного срока, возникшие по вине производителя устраняются за счет производителя.

Производитель не несет ответственности за ущерб имуществу и здоровью, нанесенный потребителю и/или третьим лицам в результате действий при монтаже, вводе в эксплуатацию и эксплуатации продукции.

Ремонт и обслуживание продукции с истекшим гарантийным сроком осуществляется за счет средств потребителя.

Доставка до места гарантийного ремонта осуществляется за счёт потребителя.

Ссылки.

1. Официальный сайт MicroPython <https://micropython.org>
2. Адаптированное под устройство руководство по языку программирования MicroPython
http://software.open-dev.ru/docs/pdf/OpenDev_MicroPython.pdf