

1С: Предприятие.

Бесплатный, сертифицированный драйвер считывателя карт

ZR1C v1.9.6

Руководство пользователя

Оглавление

ВВЕДЕНИЕ.....	7
Основные возможности.....	7
Требования к системе.....	7
БЫСТРЫЙ ОБЗОР.....	8
Установка драйвера считывателя	8
Настройка формата номера карты	8
Настройка 1С Предприятие	8
Выбор варианта работы.....	11
Классический режим работы	11
Режим работы "1С: Совместимо".....	15
Удаление компоненты.....	22
Конфигуратор ZR1C.....	22
Вкладка "Основное"	24
Вкладка "Форматы"	28
Окно "Параметры формата"	30
Окно "Фильтры ключей"	31
Окно "Параметры ключа".....	32
Окно "Настройка столбцов".....	33
Вкладка "Чтение ключей"	34
Вкладка "Общее" (Чтение ключей).....	35
Вкладка "Mifare Classic" (Чтение ключей).....	36
Вкладка "Mifare Plus SL3" (Чтение ключей)	37
Вкладка "Mifare Ultralight" (Чтение ключей)	38
Вкладка "Temic" (Чтение ключей)	39
Вкладка "Звуки"	40
Вкладка "Конфигуратор".....	41
Вкладка "Тест"	42
Окно "Подключение"	43
Вкладка "События" (Тест)	45
Вкладка "Свойства" (Тест)	46
Вкладка "Методы" (Тест)	47
Компонента ZR1C	47
Свойства компоненты	47
Свойство ИскатьСтандартныеComПорты (SearchStandardComPorts).....	49
Свойство ИскатьВиртуальныеComПорты (SearchVirtualComPorts)	50
Свойство ИскатьCCID (SearchCCID)	50
Свойство ИскатьIPПорты (SearchIPPorts).....	50
Свойство ТаймаутПоискаIP (FindIPTimeout).....	50

Свойство ПопытокПоискаIP (FindIPTries).....	50
Свойство Модель (Model).....	50
Свойство Порт (Port).....	51
Свойство Скорость (Speed).....	51
Свойство СтоповыеБиты (StopBits).....	51
Свойство ТаймаутПодключения (ConnectTimeout).....	51
Свойство ПериодПодключения (ConnectPeriod).....	52
Свойство ПериодПроверкиПорта (RxPeriod).....	52
Свойство ТаймаутЗапроса (RequestTimeout).....	52
Свойство ПопытокЗапроса (RequestTries).....	52
Свойство Виганд (Wiegand).....	52
Свойство БинарныйРежим (BinaryMode).....	52
Свойство РазныеКодыIL100 (DifferentIL100).....	52
Свойство NVT (NVT).....	53
Свойство ИДФорматаСчитывателя (ReaderFormatID).....	53
Свойство Префикс (Prefix).....	53
Свойство Суффикс (Suffix).....	54
Свойство УровеньЛога (LogLevel).....	55
Свойство МаксРазмерЛога (MaxLogSize).....	55
Свойство ТипыКлючей (KeyTypes).....	55
Свойство ТаймаутУдерживанияКлюча (KeyHoldTimeout).....	56
Свойство УдерживатьMifareC (HoldMifareC).....	56
Свойство УдерживатьMifareP (HoldMifareP).....	57
Свойство УдерживатьMifareUL (HoldMifareUL).....	57
Свойство УдерживатьTemic (HoldTemic).....	57
Свойство ПараметрыСканированияTemic (TemicScanParameters).....	57
Свойство АвтоMifareC (AutoMifareC).....	57
Свойство АвтоMifareP (AutoMifareP).....	58
Свойство АвтоMifareUL (AutoMifareUL).....	58
Свойство АвтоTemic (AutoTemic).....	58
Свойство ЗвукВключен (SoundEnabled).....	58
Свойство ЗвукПередачи (SoundTransfer).....	58
Свойство ЗвукОшибкиПередачи (SoundTransferError).....	58
Свойство ЗвукПодключения (SoundConnections).....	58
Свойство ЗвукОтключения (SoundDisconnections).....	59
Свойство ЗвукНайденКлюч (SoundFoundKey).....	59
Свойство ЗвукПотерянКлюч (SoundLostKey).....	59
Свойство КлючАвторизацииMC (AuthorizationKeyMC).....	59
Свойство НастройкиИзменены (SettingsModified).....	59

Свойство КлючАвторизацииMP (AuthorizationKeyMP)	59
Свойство КоличествоФорматов (FormatCount)	59
Свойство КоличествоФильтровКлючей (KeyFilterCount)	60
Свойство ИДФорматаПоУмолчанию (DefaultFormatID)	60
Методы компоненты	60
Метод ПолучитьНомерВерсии (GetVersion)	62
Метод ПолучитьОписание (GetDescription)	63
Метод ПолучитьОшибку (GetLastError)	63
Метод ПолучитьПараметры (GetParameters)	64
Метод УстановитьПараметр (SetParameter)	64
Метод Подключить (Open)	65
Метод Отключить (Close)	65
Метод ТестУстройства (DeviceTest)	66
Метод ПолучитьДополнительныеДействия (GetAdditionalActions)	66
Метод ВыполнитьДополнительноеДействие (DoAdditionalAction)	66
Метод ЗагрузитьНастройки (LoadSettings)	67
Метод СохранитьНастройки (SaveSettings)	67
Метод НайтиФильтрКлючейПоИД (FindKeyFilterByID)	68
Метод НайтиФильтрКлючейДляКлюча (FindKeyFilterForKey)	68
Метод ПолучитьФильтрКлючей (GetKeyFilter)	68
Метод УстановитьФильтрКлючей (SetKeyFilter)	69
Метод ВставитьФильтрКлючей (InsertKeyFilter)	70
Метод УдалитьФильтрКлючей (DeleteKeyFilter)	70
Метод ОчиститьФильтрыКлючей (ClearKeyFilters)	71
Метод НайтиФорматПоИД (FindFormatByID)	71
Метод НайтиФормат (FindFormat)	71
Метод ПолучитьФормат (GetFormat)	72
Метод УстановитьФормат (SetFormat)	72
Метод ВставитьФормат (InsertFormat)	73
Метод УдалитьФормат (DeleteFormat)	74
Метод ОчиститьФорматы (ClearFormats)	74
Метод ПроверитьФормат (CheckFormat)	74
Метод Форматировать (Format)	75
Метод НайтиСледующий (FindNext)	76
Метод ЗавершитьПоиск (FindClose)	76
Метод ОбновитьПараметрыСчитывателя (UpdateReaderParameters)	77
Метод ПолучитьСостояние (GetStatus)	77
Метод Переподключиться (Reconnect)	78
Метод ПолучитьИнфоСчитывателя (GetReaderInfo)	78

Метод ПолучитьИнфоКлюча (GetKeyInfo).....	79
Метод ПродолжитьСканирование (ContinueScanning).....	80
Метод ЧитатьMifareUL (ReadMifareUL).....	81
Метод ПисатьMifareUL (WriteMifareUL).....	85
Метод ЧитатьMifareC (ReadMifareC).....	86
Метод ПисатьMifareC (WriteMifareC).....	88
Метод ЧитатьMifareP (ReadMifareP).....	90
Метод ПисатьMifareP (WriteMifareP).....	91
Метод ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader).....	93
Метод Сигнализировать (Signal).....	94
Метод ИнициализироватьTemic (InitializeTemic).....	95
Метод ИскатьTemic (FindTemic).....	96
Метод ЧитатьTemic (ReadTemic).....	97
Метод ПисатьTemic (WriteTemic).....	98
Метод ПрерватьКоманду (BreakCommand).....	81
Метод ПолучитьСостояниеКоманды (GetCommandStatus).....	81
Метод ПолучитьРезультатКоманды (GetCommandResult).....	82
Метод ОчиститьЛог (ClearLog).....	100
Метод ДобавитьВЛог (AddLog).....	100
События компоненты.....	101
Событие ДанныеКарты (TracksData).....	101
Событие ОшибкаДрайвера (DriverError).....	102
Событие УстройствоПодключено (DeviceConnect).....	102
Событие УстройствоОтключено (DeviceDisconnect).....	103
Событие НетКарты (NoCard).....	103
Событие Команда (Command).....	103
Схема передачи номера ключа.....	105
Схема выбора формата для ключа.....	106
Коды возврата ZR1C.....	106
Формат.....	108
Шаблон.....	108
Параметры.....	110
Ключи.....	112
Mifare Classic 1K.....	112
Основные возможности Mifare Classic 1K.....	112
Организация памяти Mifare Classic 1K.....	113
Работа с памятью Mifare Classic 1K.....	114
Mifare Classic 4K.....	116
Основные возможности Mifare 4K Classic.....	116

Организация памяти Mifare Classic 4K	116
Работа с памятью Mifare Classic 4K	118
Mifare Plus.....	119
Основные возможности	120
Уровни безопасности (Security level)	120
Работа с памятью Mifare Plus	121
Mifare Ultralight	121
Основные возможности Mifare Ultralight	121
Организация памяти Mifare Ultralight	122
Temic (T5557, T5577)	123
Основные возможности Temic (T5557)	123
Организация памяти Temic (T5557)	124
Считыватели	126
Настольный считыватель Z-2 (мод. MF)/Z-2 USB MF.....	126
Настольный считыватель Z-2 (мод. MF-I)	127
Настольный считыватель Z-2 (мод. MF CCID)	127
Сетевой считыватель Matrix-III (мод. MF K Net) / Matrix-III Net	128
Сетевой считыватель CP-Z-2 (мод. MF-I).....	129
Настольный считыватель Z-2 (мод. RD_ALL)/Z-2 USB.....	129
Сетевой считыватель Matrix-III (мод. RD_All)	130
Настольный считыватель Z-2 (мод. E HTZ RF)/Z-2 EHR.....	130
Настольный считыватель Z-1 (мод. N Z)/Z-2 Base.....	131
Сетевой считыватель Matrix-V (мод. E S RF) / Matrix-V	132
Настольный считыватель Z-2 (мод. E HT Hotel)/Z-2 RF-1996.....	132
Работа с сетевой прошивкой Matrix III Rd-All.....	133
Вопросы и ответы.....	134
Пример для 1С	135
Портативный режим	136
Контакт с автором	136

ВВЕДЕНИЕ

ZR1C – это компонента, предназначенная для передачи номеров ключей от RFID-считывателя в 1С Предприятие.

Параметр	Значение
Технология создания компоненты	Native API
Тип драйвера	Подключаемый по стандарту "1С: Совместимо"
Тип оборудования	Считыватели магнитных карт
Наименование	ZR1C: Драйвер для подключения устройств ввода
Идентификатор объекта	AddIn.ZR1CExtension
Имя файла драйвера	ZR1C_<версия>.zip
Версия драйвера	<версия zr1c.dll>

Комплектация:

1. ZR1C.dll, ZR1C64.dll – 32- и 64- битные версии компоненты;
2. manifest.xml – файл, описывающий содержимое ZIP-архива ("ZR1C_<версия>.zip") для загрузки в конфигурацию 1С;
3. ConfigZR1C.exe, ConfigZR1C64.exe – 32- и 64- битные версии утилиты "Конфигуратор ZR1C" для настройки компоненты;
4. ZR1C_RUS.chm – данный файл описания компоненты.

Примечание

Для импорта zip файла в ПО "1С: Предприятие" нужно удалить из архива все файлы, кроме: ZR1C.dll, ZR1C64.dll, manifest.xml и info.xml.

Основные возможности

- Чтение номеров карт с помощью RFID-считывателя IronLogic;
- Получение [событий](#) о поднесении/удалении карты;
- Гибкая [настройка формата](#) передаваемого номера для разных типов карт;
- Авто чтение номера из памяти ключа (страница [Чтение ключей](#)):
 - [Mifare Classic](#) (считыватели: [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#));
 - [Mifare Plus SL3](#) (считыватели: [Z-2 MF-I](#));
 - [Mifare Ultralight](#) (считыватели: [Z-2 RD_ALL](#), [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Rd-All](#), [Matrix III Net](#), [CP-Z2-MF](#));
 - [Temic](#) (считыватели: [Z-2 RD_ALL](#), [Z-2 EHR](#));
- Возможность чтения/записи памяти ключей:
 - [Mifare Classic](#) (методы [ReadMifareC](#), [WriteMifareC](#));
 - [Mifare Plus SL3](#) (методы [ReadMifareP](#), [WriteMifareP](#));
 - [Mifare Ultralight](#) (методы [ReadMifareUL](#), [WriteMifareUL](#));
 - [Temic](#) (методы [ReadTemic](#), [WriteTemic](#));
- Работа с сетевой прошивкой [Matrix III Rd-All](#);
- Звуковые уведомления (о подключении/отключении считывателя, о поднесении и удалении карты, о передаче номера в 1С, об ошибке).

Требования к системе

ОС: Windows® XP SP2 /Vista/7/8/10 (32- или 64- битная версия)

Материнская плата: один USB-порт если считыватель подключается по usb

Место на диске: ~4 Мб

Требования к RFID-считывателю:

Поддерживаемые модели:

- подключаемые по USB: Z-2 USB, Z-2 Rd All, Z-2 USB MF, Z-2 EHR, Z-2 Base, RF-1996, Z-2 MF-I, Z-2 MF CCID
- подключаемые через конвертер (USB или IP): Matrix III Rd-All, Matrix III Net, Matrix V
 - поддерживаемые модели конвертеров: Z-397, Z-397 Guard (в режиме Normal), Z-397 Web (режим Server), Z-397 IP (режим Server)

Прошивки считывателей: только заводские версии

Драйвер: для моделей считывателей и конвертеров, подключаемых по USB, нужен драйвер (доступен на сайте www.ironlogic.ru)

Требования к 1С Предприятие:

Версия: 8.3

Тип: Толстый клиент (обычное приложение)

БЫСТРЫЙ ОБЗОР

Установка драйвера считывателя

Для считывателей, которые подключаются по USB, нужно установить драйвер. После установки драйвера появится виртуальный COM-порт, через который ZR1C подключается к считывателю. Драйвер для USB считывателя доступен на сайте ironlogic.ru на странице считывателя. Инструкция по установке драйвера находится в архиве с драйвером.

Настройка формата номера карты

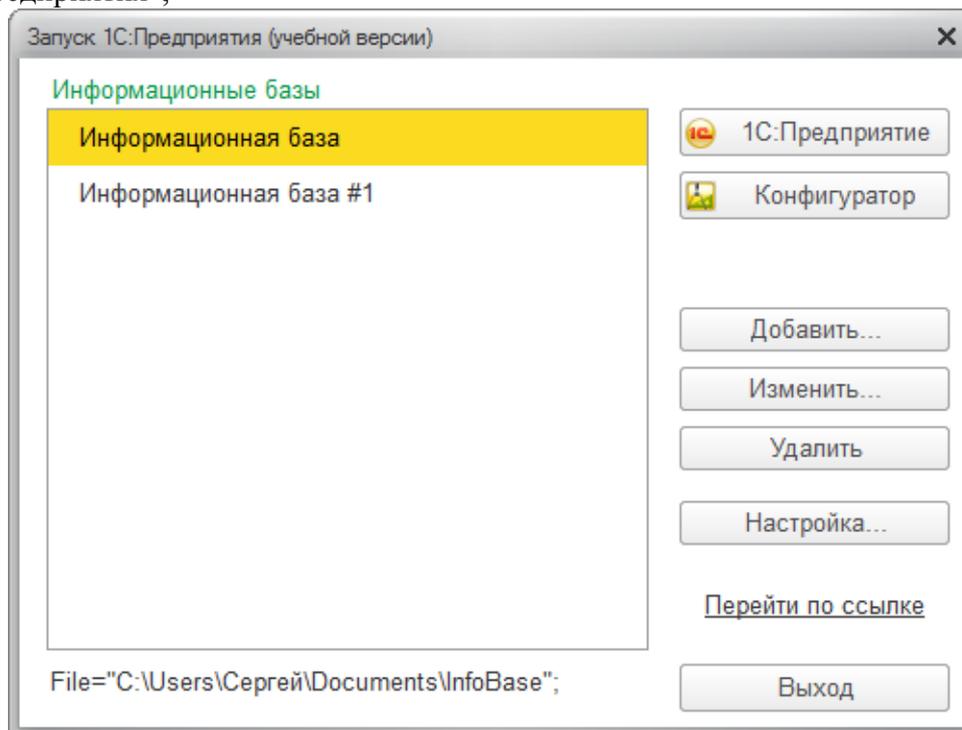
Настроить формат передаваемых номеров карты можно в "ConfigZR1C.exe":

1. Запустите ConfigZR1C.exe;
2. На вкладке "Основное" в группе "Настройки компоненты ZR1C" установите флаг "Загрузить при старте ZR1C";
3. На вкладке "Форматы" выберите формат (подробнее см. в [описании ConfigZR1C](#));
4. Нажмите кнопку "Сохранить";
5. Перезапустите 1С.

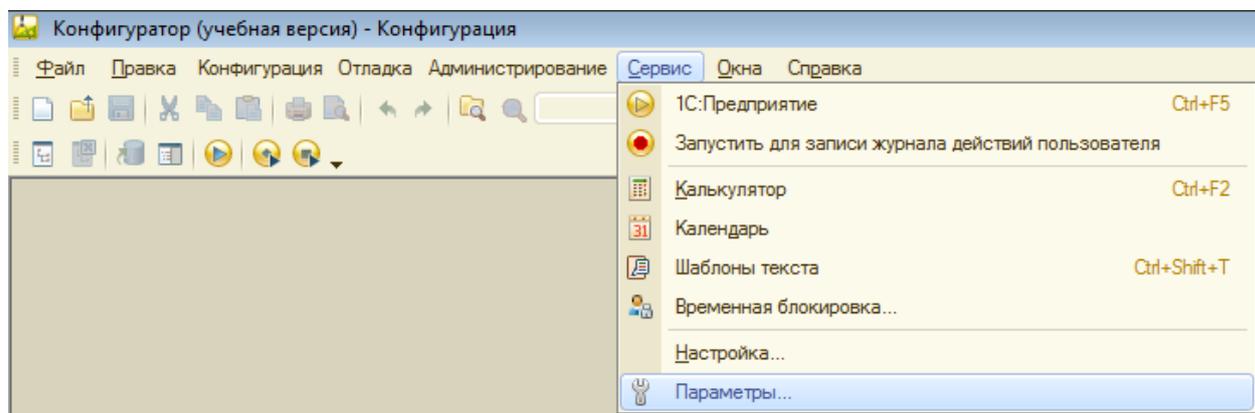
Настройка 1С Предприятие

Включение режима "Толстый клиент (обычное приложение):

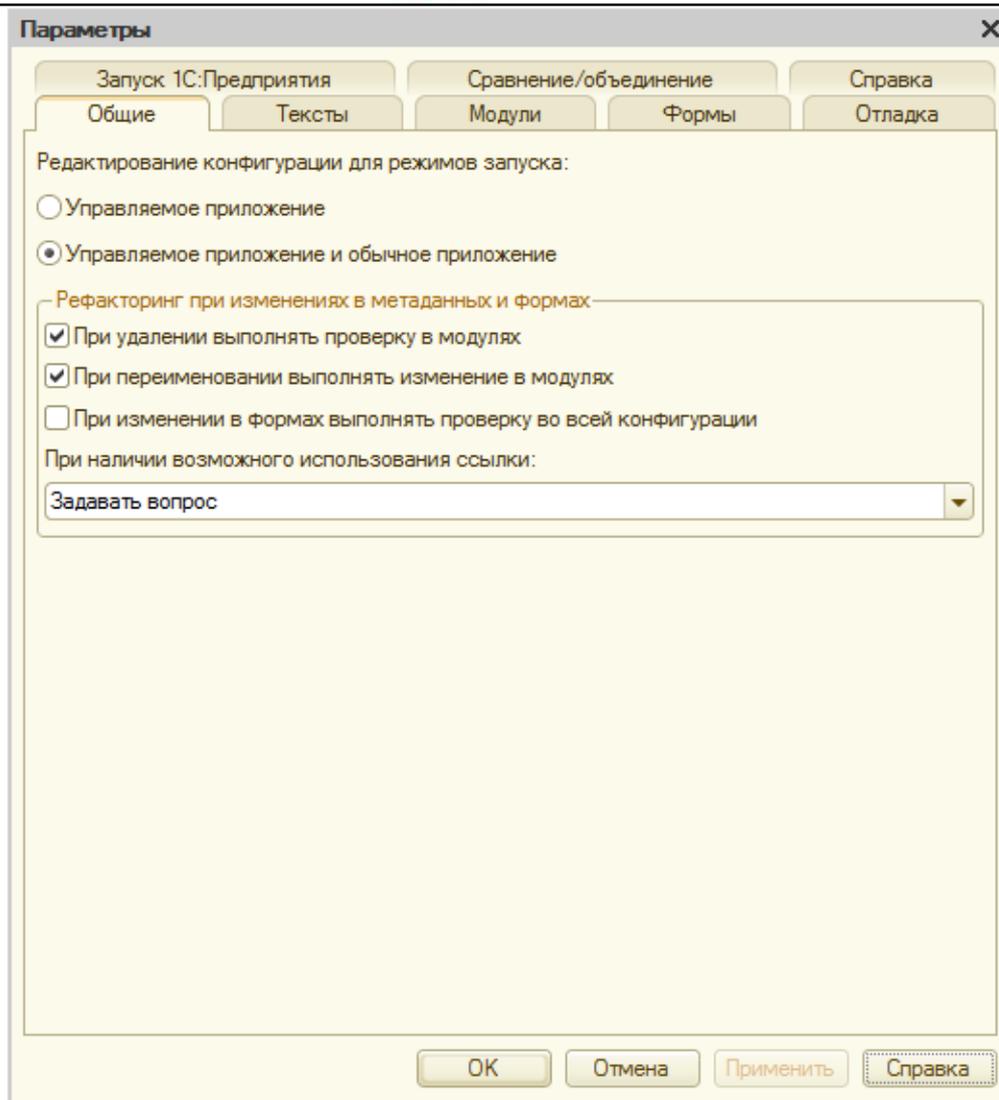
1. Запустите ярлык на рабочем столе "1С Предприятие". Появится окно "Запуск 1С: Предприятия";



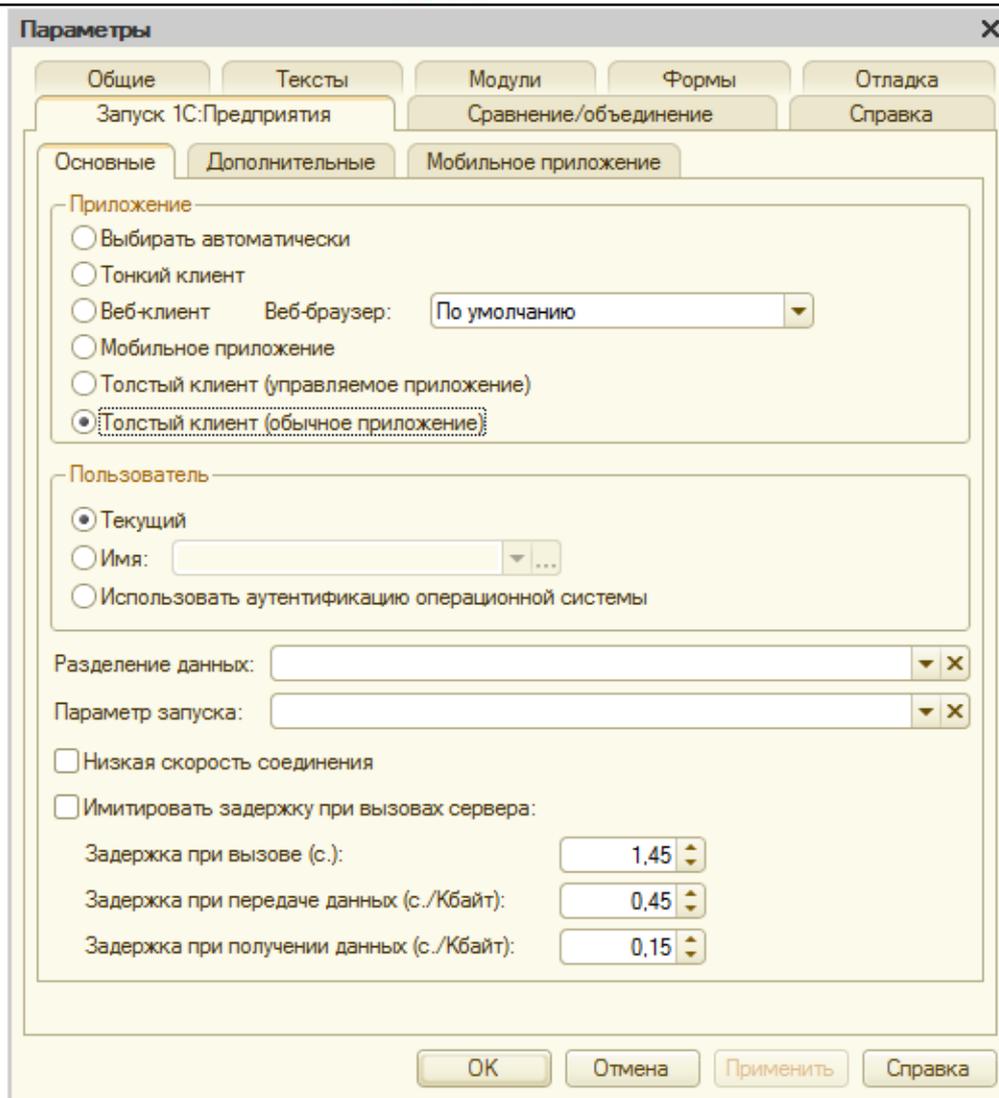
2. В списке "Информационные базы" выберите базу;
3. Нажмите кнопку "Конфигуратор". Появится окно "Конфигуратор";
4. В меню "Сервис" выберите команду "Параметры...".



Появится окно "Параметры":



5. Кликните по вкладке "Запуск 1С: Предприятия".



6. В группе "Приложение" выберите "Толстый клиент (обычное приложение)";
7. Нажмите кнопку "ОК".

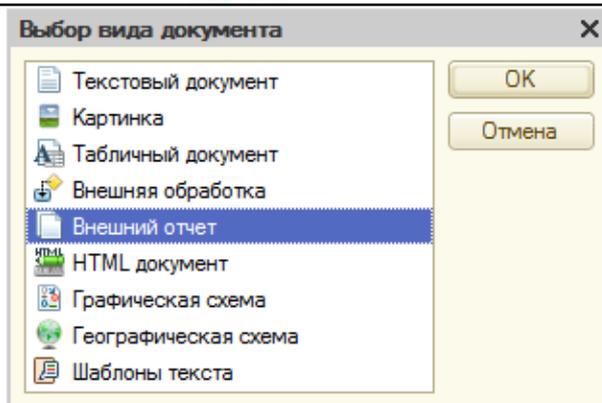
Выбор варианта работы

С компонентой можно работать 2 способами:

- [Классический](#) – нужно писать код на языке 1с: в обработчиках формы подключаться к считывателю, и ловить номера карт, подносимых к считывателю;
- ["1С: Совместимо"](#) – использовать интерфейс надстройки "Библиотека подключаемого оборудования": выбрать считыватель, нажать кнопку "Подключить", и ловить номера карт, подносимых к считывателю.

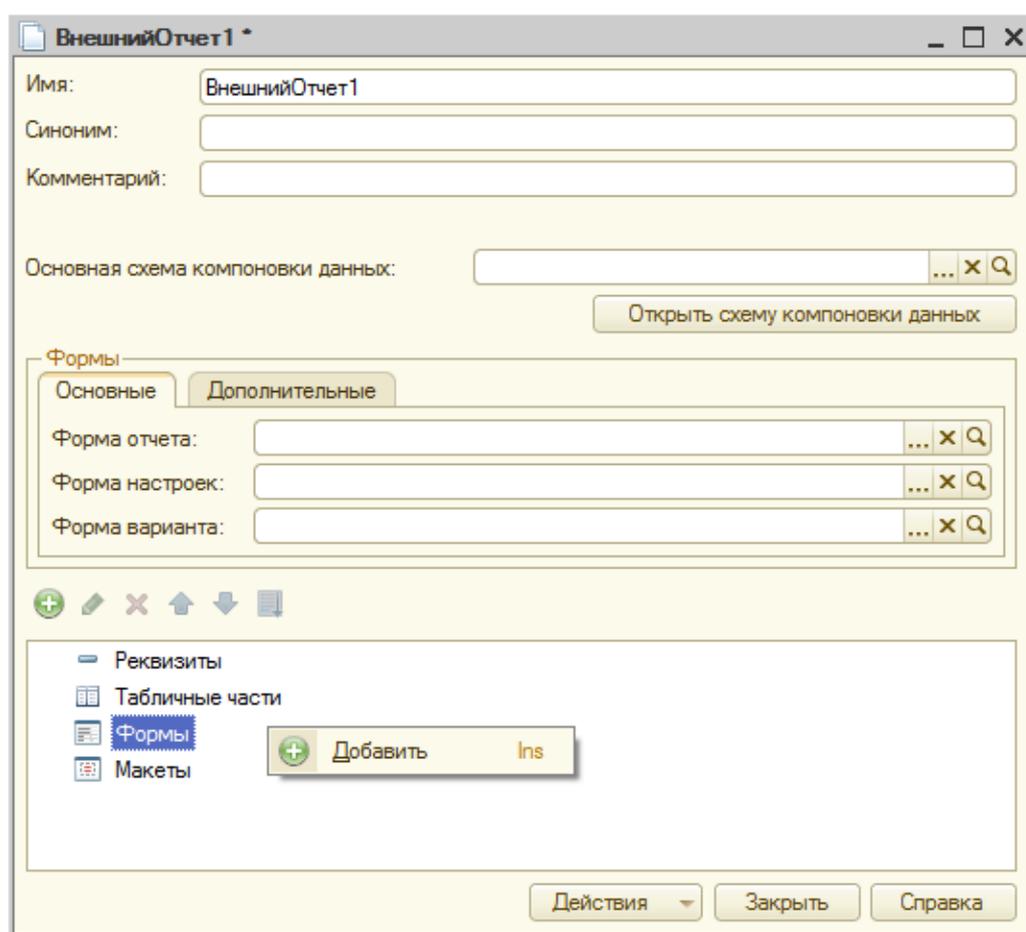
Классический режим работы

1. Запустите "1С Предприятие". Появится окно "Запуск 1С: Предприятие (учебной версии)";
2. Выберите базу и нажмите кнопку "Конфигуратор";
3. Откройте (или создайте) обычную форму. Например:
 1. В меню выберите "Файл" -> "Новый...". Появится окно "Выбор вида документа".

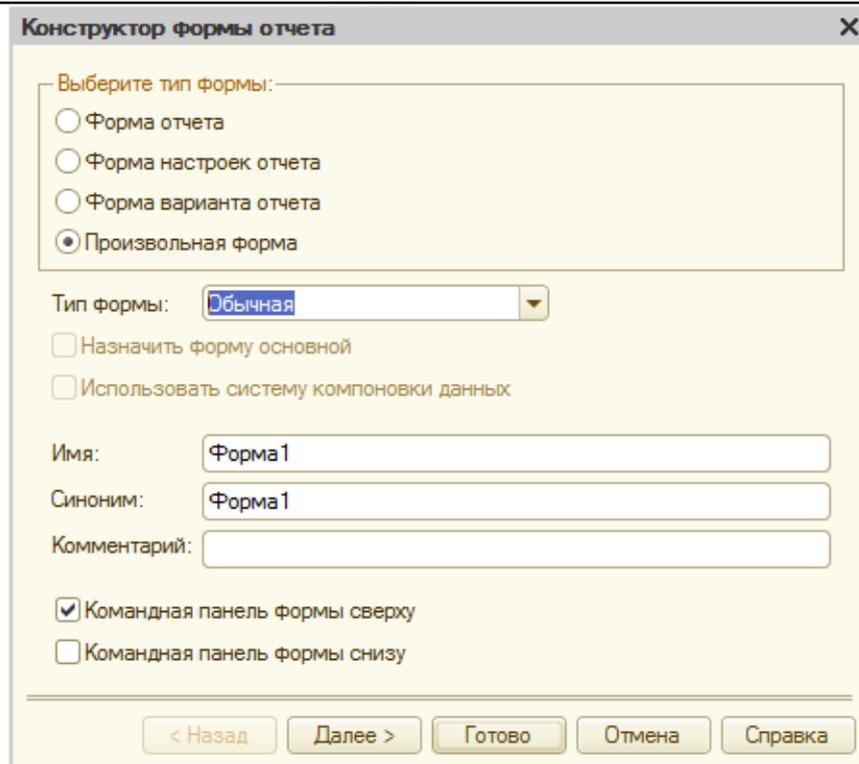


2. Выберите "Внешний отчет";

Нажмите кнопку "OK". Появится окно "ВнешнийОтчет1";

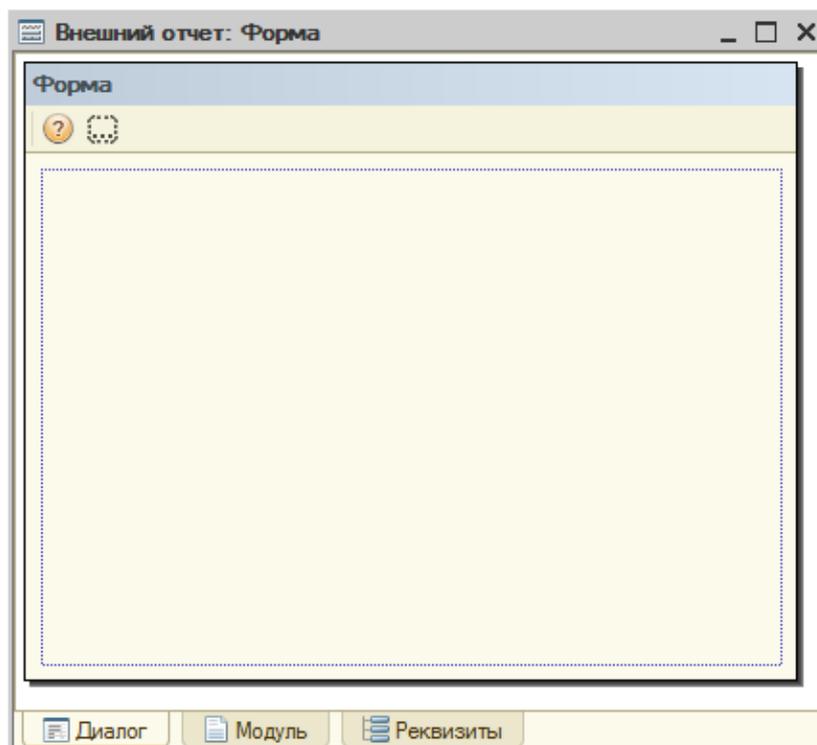


3. Кликнете ПКМ по "Формы" и выберите "Добавить";

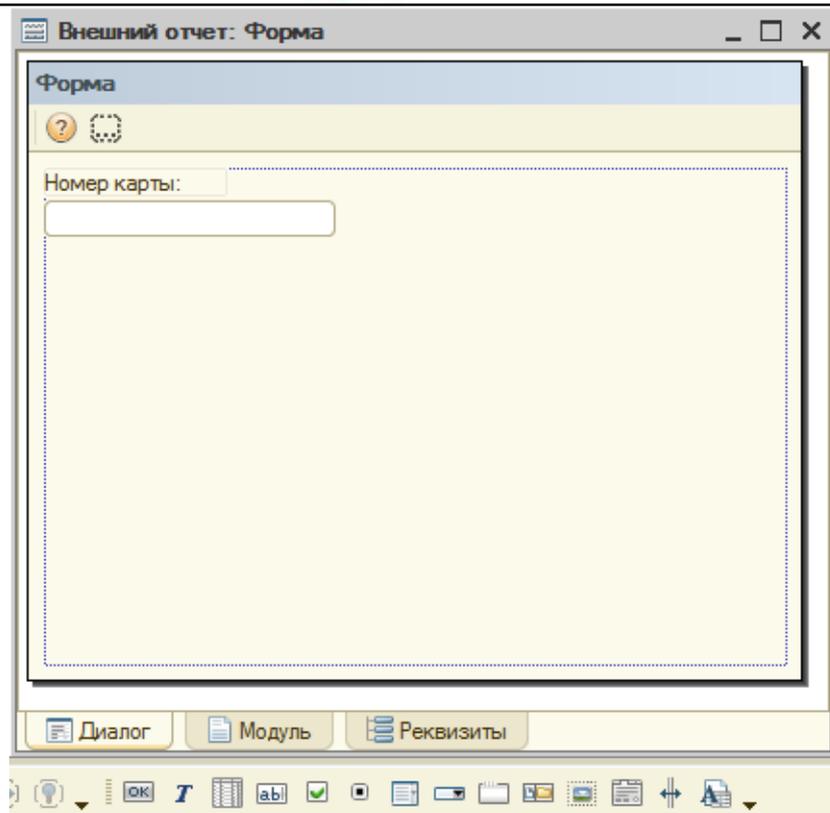


4. В группе "Выберите тип формы" выберите "Произвольная форма";
5. Выберите тип формы "Обычная";

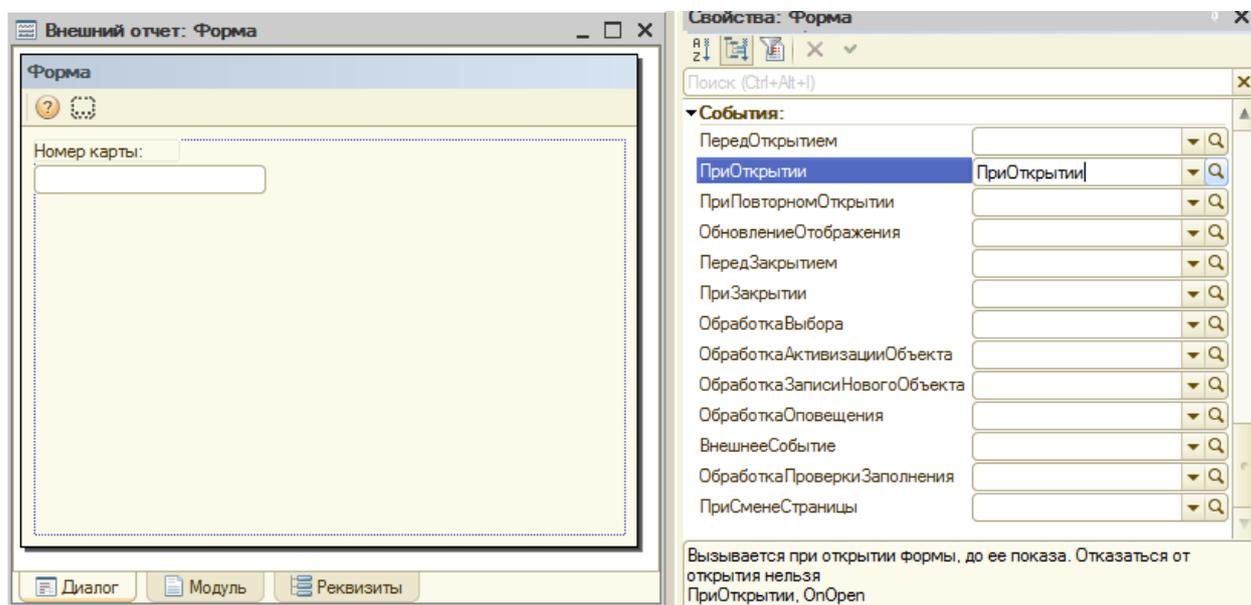
Нажмите "Готово". Появится форма;



6. С помощью набора инструментов под формой вставьте на форму 2 элемента управления: 1) Надпись "Номер карты:", 2) Поле ввода.



7. Кликните по свободному пространству формы, и затем в панели "Свойства: Форма" (находится справа от формы) выберите параметр "ПриОткрытии" и нажмите кнопку  (или сочетание клавиш Ctrl+Shift+F4).



4. Аналогично создайте обработчик события "ВнешнееСобытие";
 5. После создания обработчика появится окно редактора кода обработчика (вкладка "Модуль"), в нём напишите код:

```

Перем Компонента;
Перем DevId;
Перем СчитывательПодключен;

Процедура ПриОткрытии()

```

```
// Подключаем компоненту
ПодключитьВнешнююКомпоненту("C:\1C\ZR1C.dll", "Comp",
ТипВнешнейКомпоненты.Native);
Компонента = Новый("AddIn.Comp.ZR1CExtension");

// Задаём имя порта считывателя
Компонента.УстановитьПараметр("Порт", "Авто");

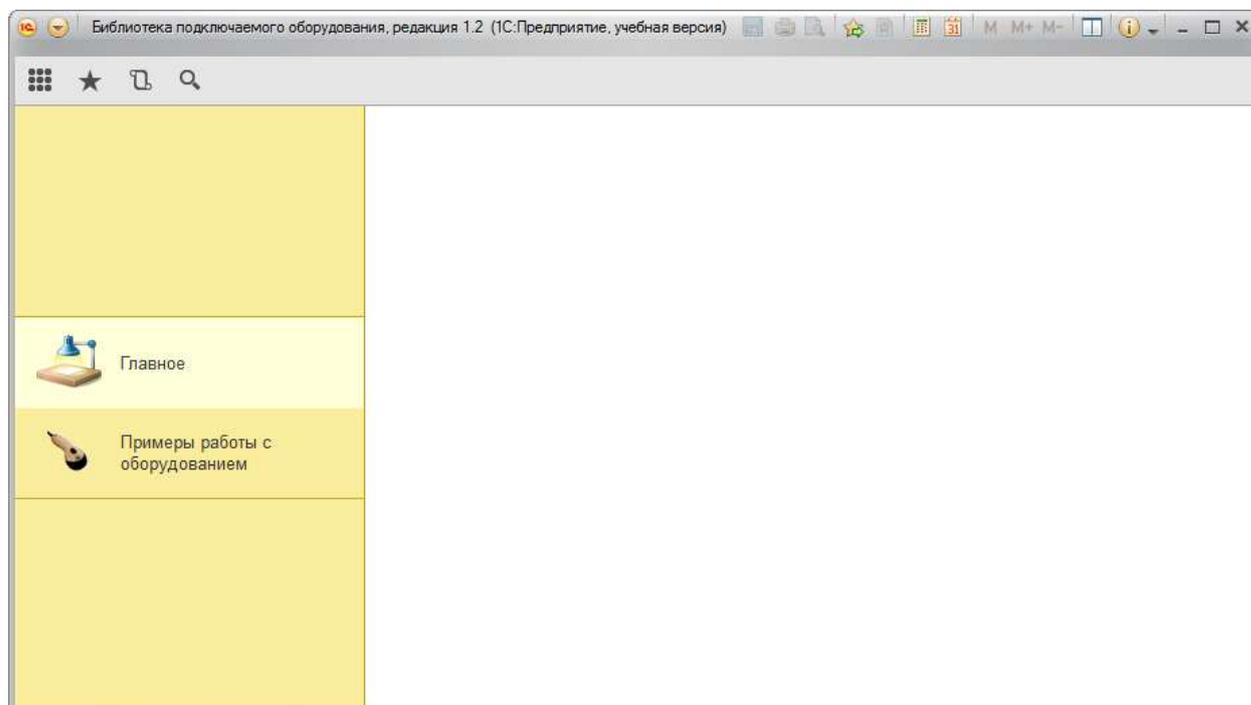
// Подключаемся к считывателю, в DevId получаем имя порта
СчитывательПодключен = Компонента.Подключить(DevId);
КонецПроцедуры

Процедура ВнешнееСобытие(Источник, Событие, Данные)
// Копируем номер карты в поле ввода
ПолеВвода1 = Данные;
КонецПроцедуры
```

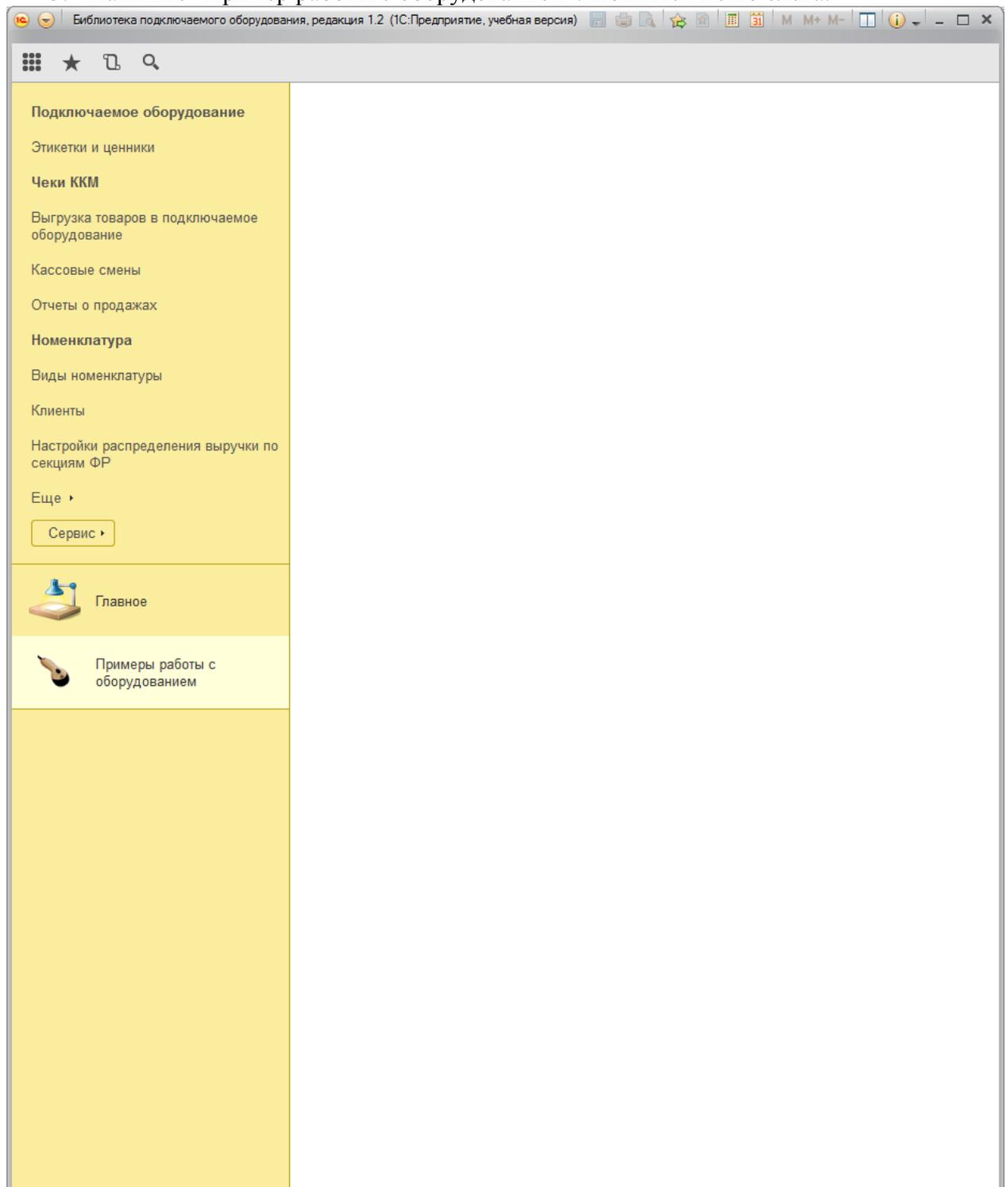
6. Сохраните файл (Ctrl+S);
7. Запустите отладку (F5). Появится окно "1С: Предприятие - Конфигурация";
8. Откройте файл (Ctrl+O). Появится окно формы;
9. Поднесите карту к считывателю. В поле "Номер карты:" появится её номер.

Режим работы "1С: Совместимо"

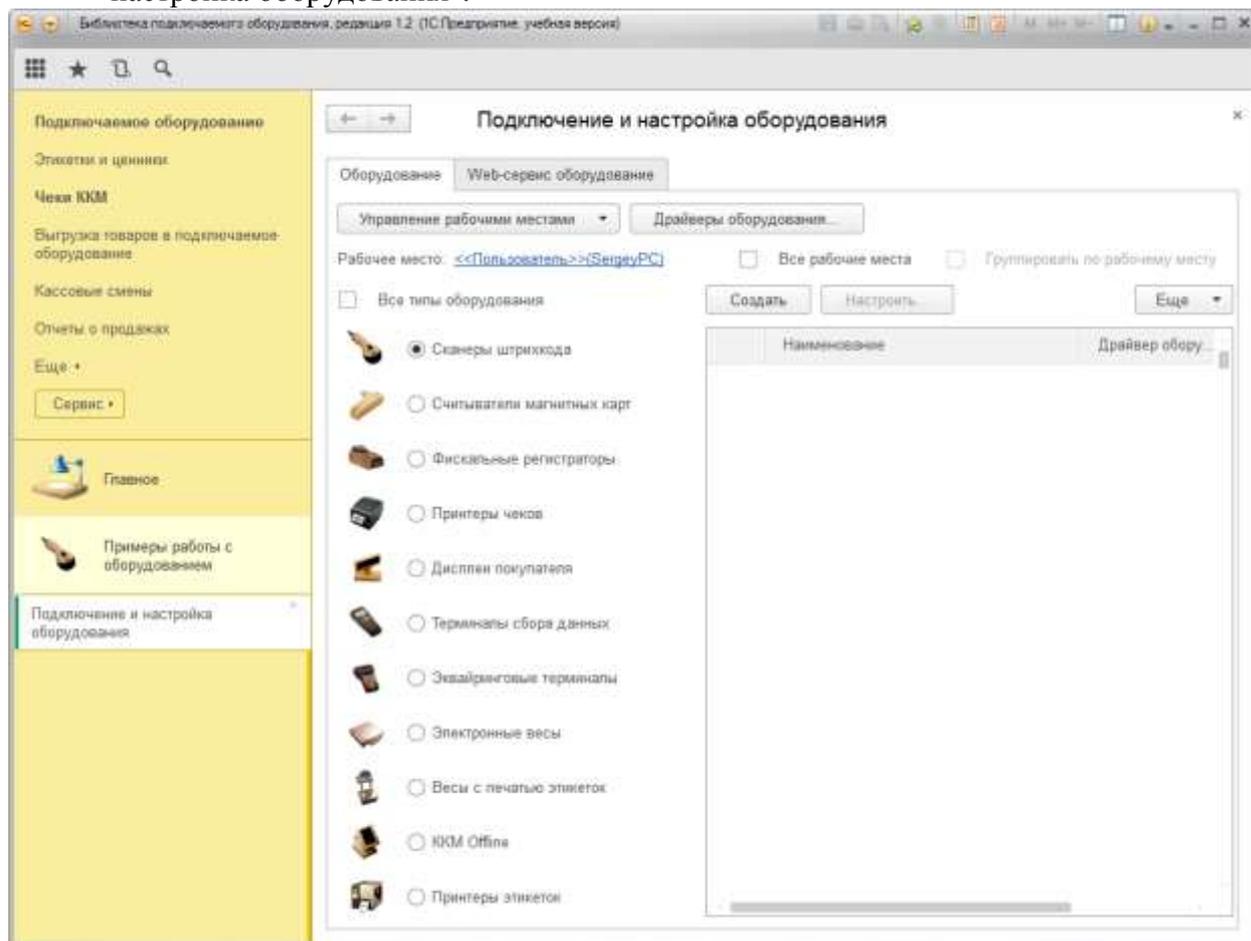
1. Запустите "1С Предприятие". Появится окно "Запуск 1С: Предприятие (учебной версии)";
2. Выберите базу и нажмите кнопку "1С: Предприятие". Появится окно "Библиотека подключаемого оборудования. редакция 1.2 (1С: Предприятие, учебная версия);



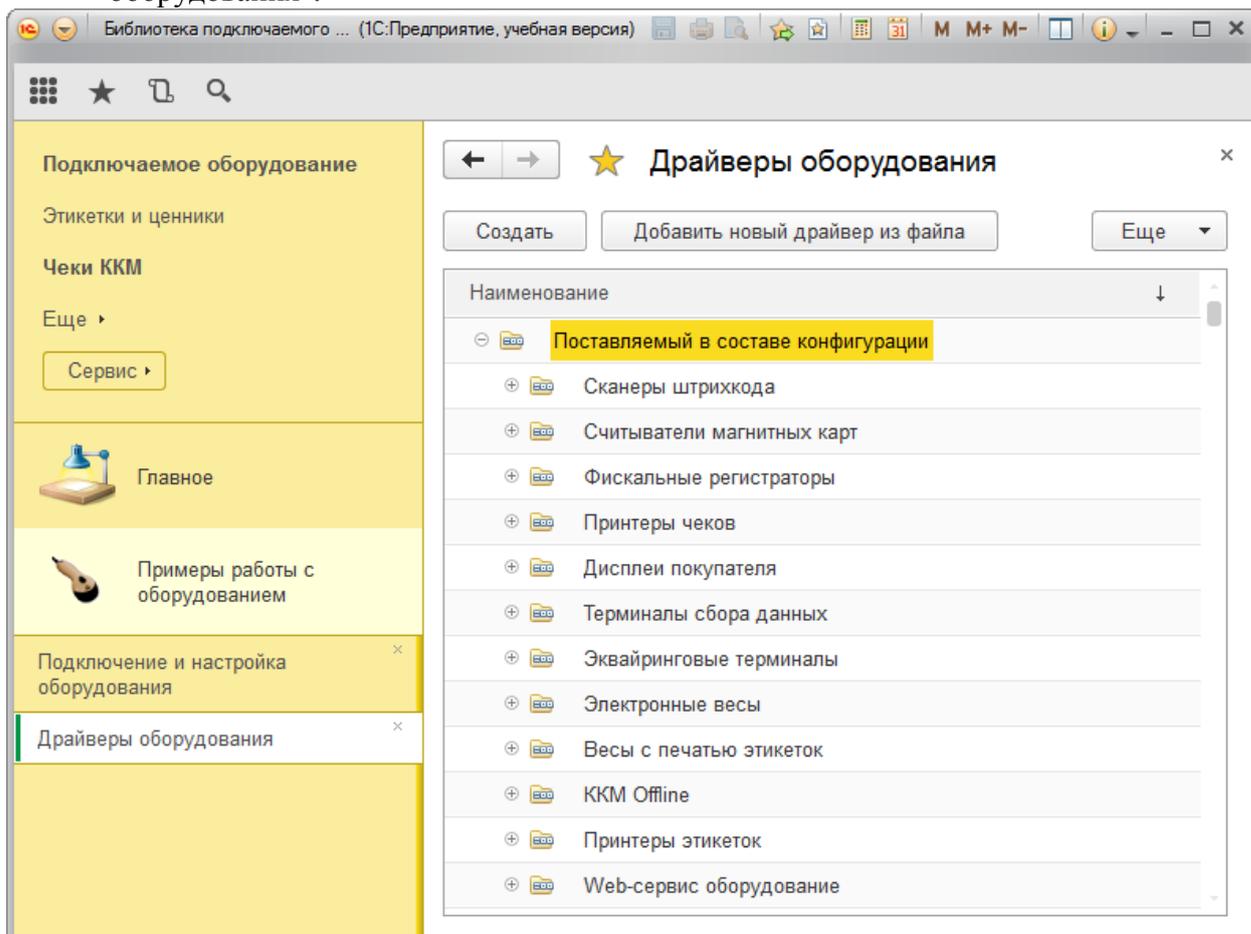
3. Нажмите "Пример работы с оборудованием". Появится меню слева.



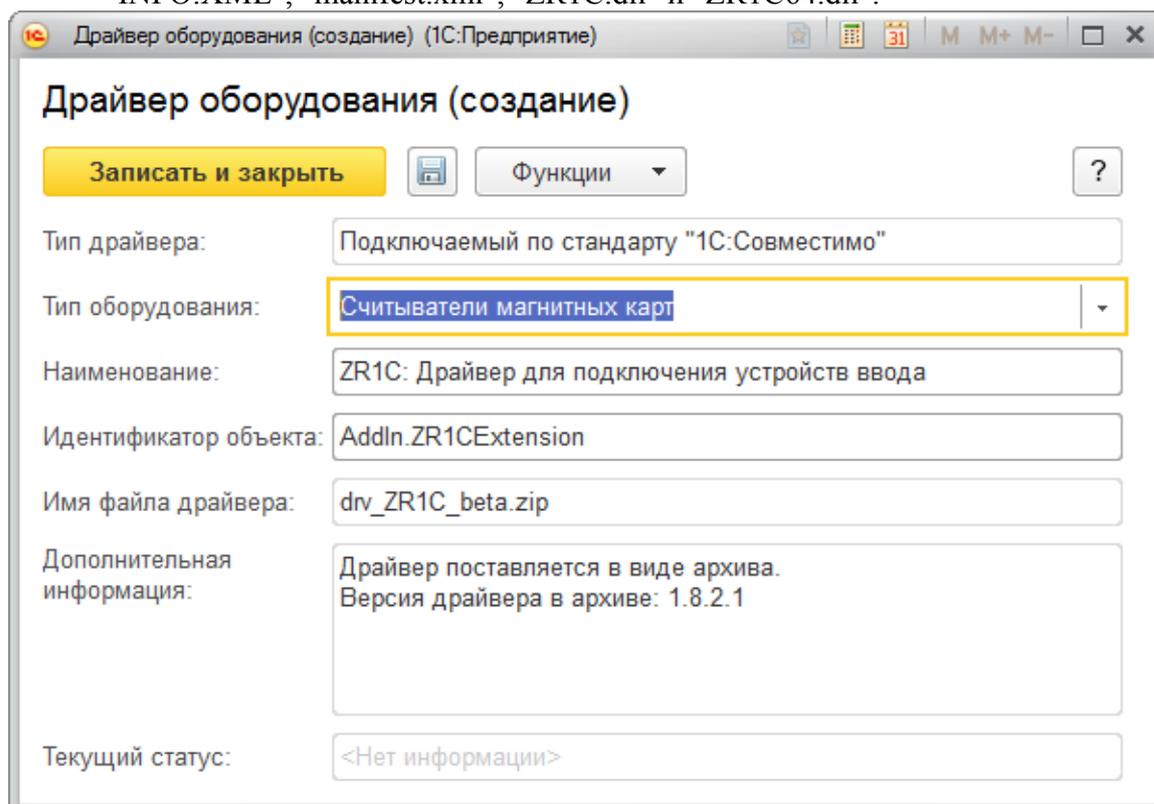
4. Нажмите "Подключаемое оборудование". Появится страница "Подключение и настройка оборудования".



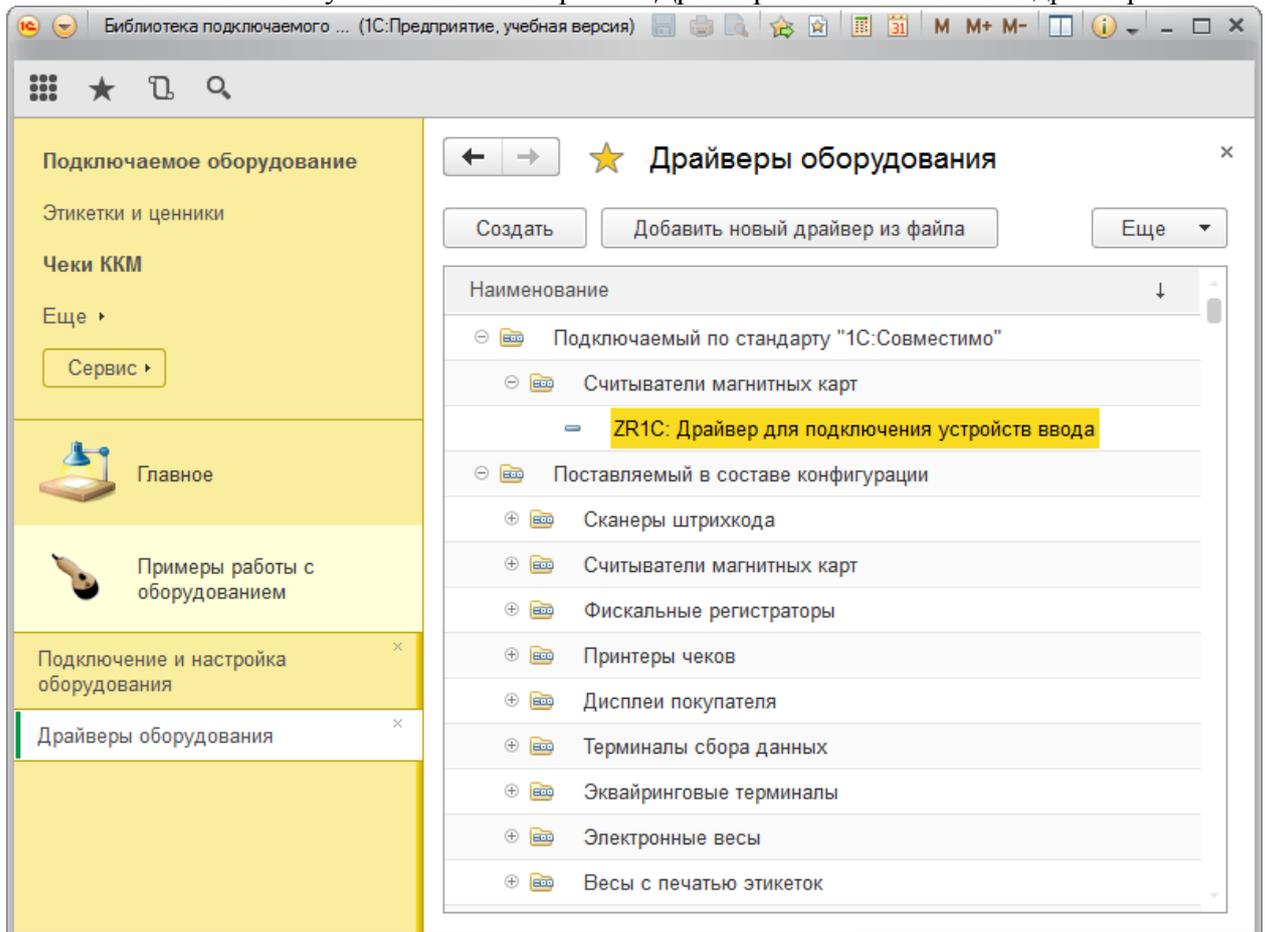
5. Нажмите кнопку "Драйверы оборудования...". Появится страница "Драйверы оборудования".



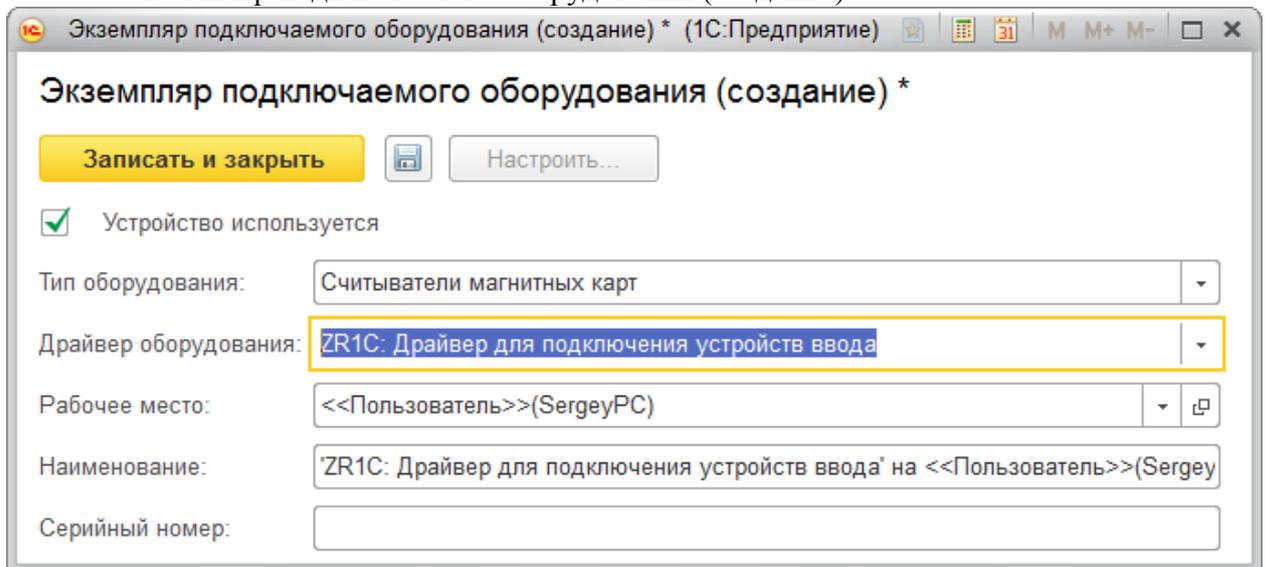
6. Нажмите кнопку "Добавить новый драйвер из файла", выберите файл "drv_ZR1C_beta.zip". Появится окно "Драйвер оборудования (создание)". Предварительно нужно удалить из архива "drv_ZR1C_beta.zip" все файлы, кроме: "INFO.XML", "manifest.xml", "ZR1C.dll" и "ZR1C64.dll".



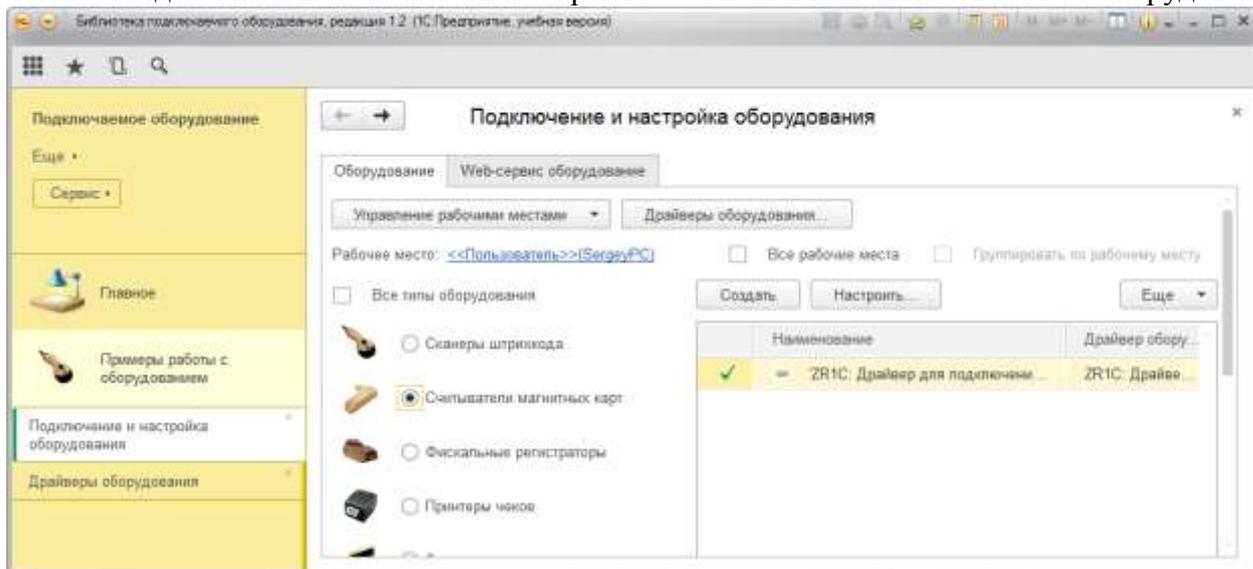
7. Нажмите кнопку "Записать и закрыть". Драйвер появится в списке драйверов.



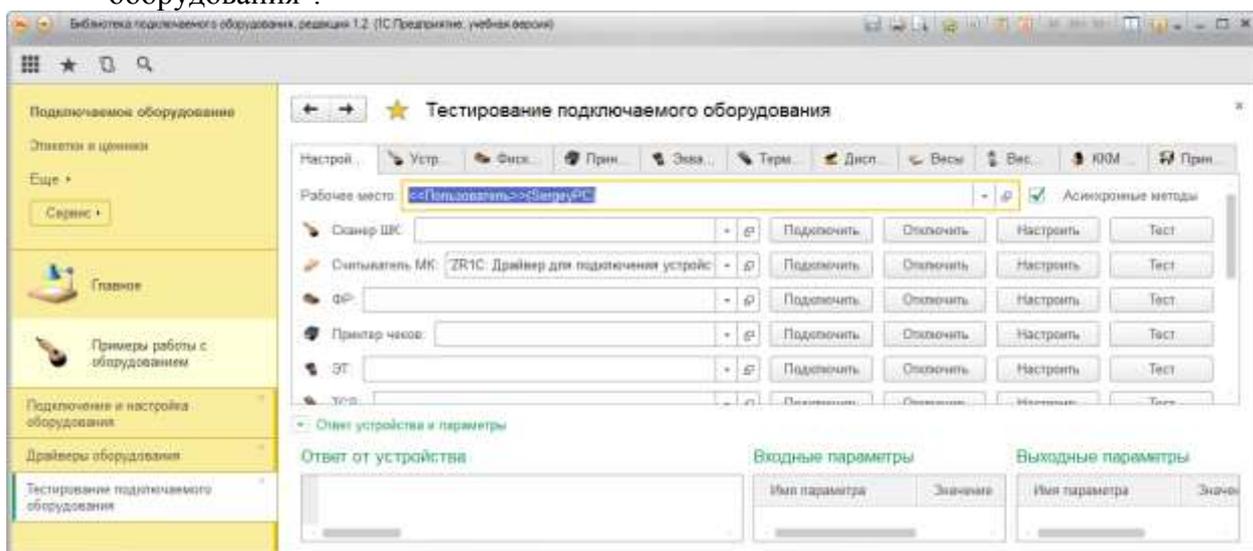
8. Нажмите "Подключение и настройка оборудования" (слева внизу), выберите "Считыватели магнитных карт", нажмите кнопку "Создать". Появится окно "Экземпляр подключаемого оборудования (создание)".



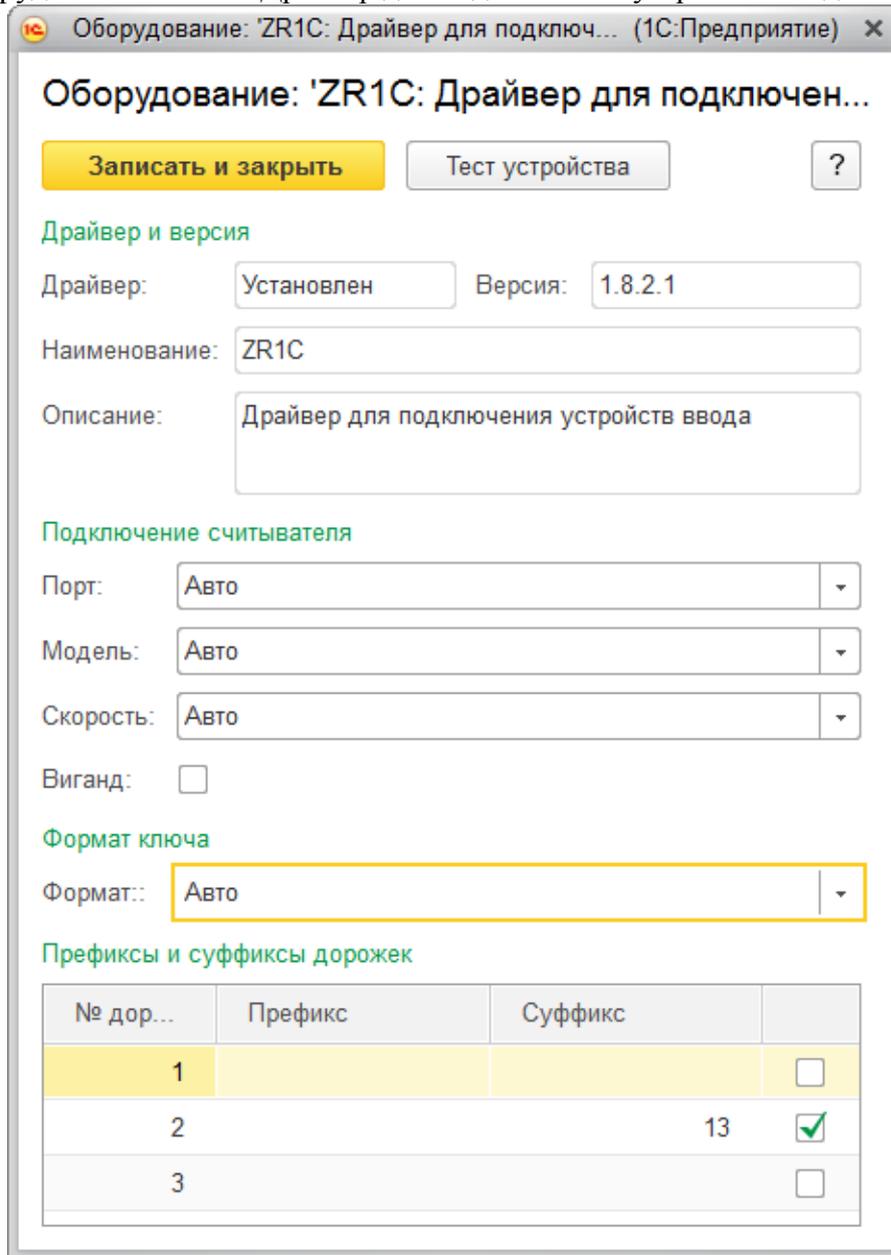
9. В поле "Драйвер оборудования" выберите "ZR1C: Драйвер для подключения устройств ввода" и нажмите "Записать и закрыть". Компонента появится в списке оборудования.



10. Нажмите кнопку "Сервис" (слева) и в выпавшем меню – команду "Тестирование подключаемого оборудования". Появится страница "Тестирование подключаемого оборудования".



11. В строке "Считыватель МК" нажмите кнопку "Настроить". Появится окно "Оборудование: 'ZR1C: Драйвер для подключения устройств ввода'...".



Оборудование: 'ZR1C: Драйвер для подключ...' (1С:Предприятие) X

Оборудование: 'ZR1C: Драйвер для подключен...

Записать и закрыть Тест устройства ?

Драйвер и версия

Драйвер: Версия:

Наименование:

Описание:

Подключение считывателя

Порт: ▾

Модель: ▾

Скорость: ▾

Виганд:

Формат ключа

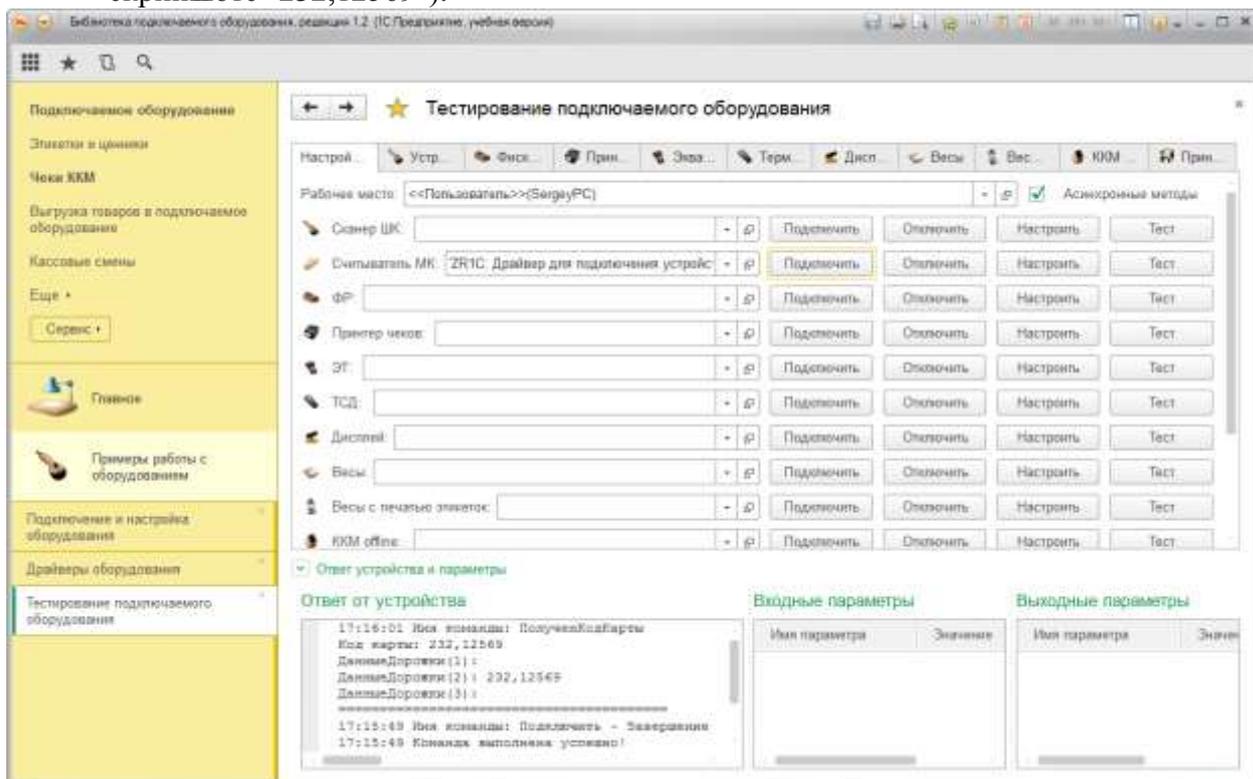
Формат: ▾

Префиксы и суффиксы дорожек

№ дор...	Префикс	Суффикс	
1			<input type="checkbox"/>
2		13	<input checked="" type="checkbox"/>
3			<input type="checkbox"/>

12. Нажмите "Записать и закрыть";
13. В строке "Считыватель МК" нажмите кнопку "Подключить", в поле "Ответ от устройства" появится "Команда выполнена успешно!";

14. Поднесите карту к считывателю, в поле "Ответ от устройства" появится её номер (на скриншоте "232,12569").



15. Готово.

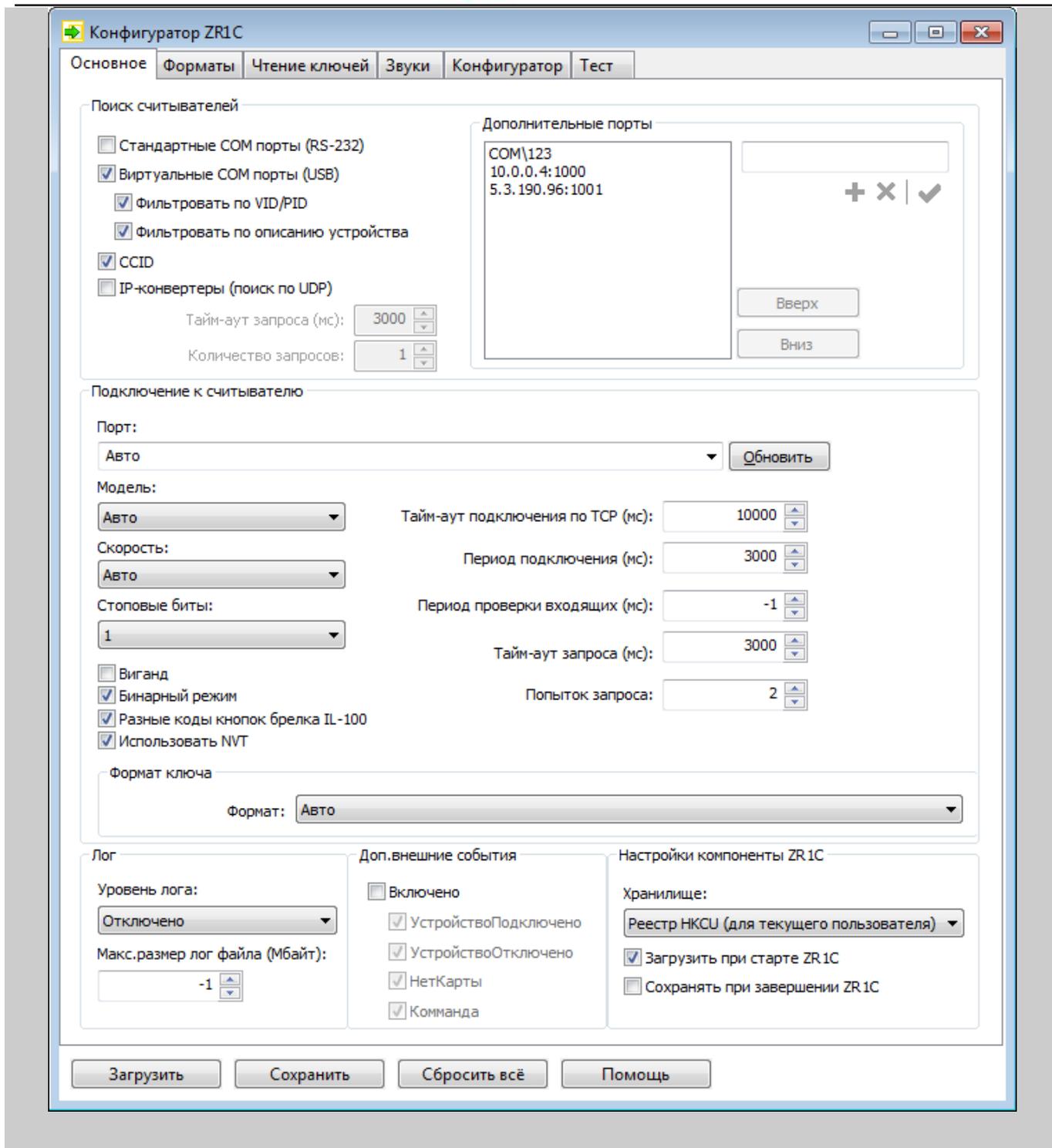
Удаление компоненты

Чтобы удалить драйвер нужно:

1. Отключиться от считывателя: на странице "Тестирование подключаемого оборудования" нажать "Отключить";
2. Удалить объект оборудования: на странице "Подключение и настройка оборудования" кликнуть ПКМ по "ZR1C: Драйвер подключаемого оборудования" и выбрать "Пометить на удаление";
3. Удалить драйвер оборудования: на странице "Драйверы оборудования" кликнуть ПКМ по "ZR1C: Драйвер для подключения устройств ввода" и выбрать "Пометить на удаление";
4. Завершить удаление: в левом верхнем углу нажать на кнопку со стрелкой "вниз" и выпавшем меню выбрать "Все функции...". Появится окно "Все функции";
5. В группе "Стандартные" дв.кликнуть по "Удаление помеченных объектов";
6. Нажать кнопку "Удалить".

Конфигуратор ZR1C

Конфигуратор ZR1C (ConfigZr1c.exe) предназначен для настройки компоненты ZR1C (zr1c.dll). После запуска **Конфигуратор ZR1C** на экране появится главное окно программы.



Вкладка "[Основное](#)" предназначена для настройки подключения к считывателю.

Вкладка "[Форматы](#)" предназначена для настройки форматов номеров ключей, передаваемых от считывателя к IC.

Вкладка "[Чтение ключей](#)" предназначена для настройки чтения ключей.

Вкладка "[Звуки](#)" предназначена для настройки звуковых уведомлений компоненты ZR1C.

Вкладка "[Конфигуратор](#)" предназначена для настройки подтверждений в программе "Конфигуратор ZR1C".

Вкладка "[Тест](#)" предназначена для тестирования компоненты "ZR1C.dll" без использования IC.

Кнопки "**Загрузить**" и "**Сохранить**" – загружает/сохраняет настройки компоненты (параметры подключения к считывателю и список включенных форматов) из реестра и настройки конфигулятора (список форматов и список групп типов ключей).

Кнопка "**Сбросить всё**" – сбрасывает все настройки на значения по умолчанию.

Кнопка "**Помощь**" – показывает этот файл помощи.

Вкладка "Основное"

Конфигуратор ZR1C

Основное | Форматы | Чтение ключей | Звуки | Конфигуратор | Тест

Поиск считывателей

- Стандартные COM порты (RS-232)
- Виртуальные COM порты (USB)
 - Фильтровать по VID/PID
 - Фильтровать по описанию устройства
- CCID
- IP-конвертеры (поиск по UDP)

Тайм-аут запроса (мс): 3000

Количество запросов: 1

Дополнительные порты

COM\123
10.0.0.4:1000
5.3.190.96:1001

+ × | ✓

Вверх
Вниз

Подключение к считывателю

Порт: Авто [Обновить]

Модель: Авто

Скорость: Авто

Стоповые биты: 1

Виганд

Бинарный режим

Разные коды кнопок брелка IL-100

Использовать NVT

Формат ключа: Авто

Тайм-аут подключения по TCP (мс): 10000

Период подключения (мс): 3000

Период проверки входящих (мс): -1

Тайм-аут запроса (мс): 3000

Попыток запроса: 2

Лог

Уровень лога: Отключено

Макс. размер лог файла (Мбайт): -1

Доп. внешние события

Включено

УстройствоПодключено

УстройствоОтключено

НетКарты

Комманда

Настройки компоненты ZR1C

Хранилище: Реестр НКCU (для текущего пользователя)

Загрузить при старте ZR1C

Сохранять при завершении ZR1C

Загрузить | Сохранить | Сбросить всё | Помощь

Эта страница предназначена для настройки подключения к считывателю.

Группа "Поиск считывателей"

Стандартные COM порты (RS-232)

Если включен, то разрешает опрос считывателей IronLogic через стандартные COM порты. По умолчанию выключено.

Виртуальные COM порты (USB)

Если включен, то разрешает опрос USB-считывателей IronLogic через виртуальные COM-порты. По умолчанию включено.

Фильтровать по VID/PID

Если включен, то исключает виртуальные COM-порты, у которых VID или PID считывателя не является VID/PID IronLogic. По умолчанию включено.

Фильтровать по описанию устройства

Если включен, то исключает виртуальные COM порты, у которых описание USB-считывателя не является описанием считывателя IronLogic (в Диспетчере устройств описание можно увидеть в окне "Свойства" для USB-считывателя -> на вкладке "Сведения" -> значение свойства "Описание устройства, предоставленное шиной"). По умолчанию включено.

CCID

Если включен, то ищет считыватели Z-2 MF CCID. По умолчанию включено.

IP-конвертеры (поиск по UDP)

Если включен, то разрешает опрос считывателей, подключенных через ip-конвертеры (UDP 9000). По умолчанию выключено;

Тайм-аут запроса

Тайм-аут запроса поиска конвертера в миллисекундах. По умолчанию 3000.

Количество запросов

Количество попыток запроса. По умолчанию 1.

Избранные порты

Предназначено для настройки списка вариантов портов в окне «Подключение» (в ПО "IC: Предприятие"), которые компонента сама обнаружить не может. Например, здесь могут быть адреса считывателей с [сетевой прошивкой Matrix III Rd-All](#), или адреса IP конвертеров. Если список пуст, то в окне "Подключение" в список поля "Порт" добавляются порты: COM1 - COM64.

- Поле ввода – позволяет вводить имя порта;
- Кнопка  – добавляет в список введенное имя порта;
- Кнопка  – удаляет выделенный в списке порт;
- Кнопка  – применяет введенное имя порта к выделенному в списке порту.

Группа "Подключение к считывателю"

Порт

Имя порта считывателя (COM-порт или адрес порта IP-конвертера, к которому подключен считыватель). Если "Auto", то используется первый найденный порт считывателя.

Если используется сетевая прошивка Matrix III Rd-All, то имя порта должно быть в формате "COM3\123", где "COM3" – порт конвертера, "123" – серийный номер (с/н) считывателя. По умолчанию "Auto";

Кнопка "Обновить" – обновляет список COM портов.

Модель

Модель считывателя. Если "Авто", то определяется автоматически. Выбор определенной модели (не "Auto") может ускорить подключение к считывателю. По умолчанию "Авто".

Скорость

Предельная скорость считывателя. Если выбрано "Авто", то определяется автоматически (максимальная из поддерживаемых считывателем). Позволяет ограничить скорость работы со считывателем. По умолчанию "Авто".

Стоповые биты

Количество стоповых бит для подключения через COM-порт. По умолчанию 1.

Тайм-аут подключения по TSP

Тайм-аут подключения к IP-конвертеру по TSP в миллисекундах. Если -1, то используется системное значение тайм-аута. По умолчанию 10000.

Тайм-аут запроса

Тайм-аут запроса к считывателю в миллисекундах. По умолчанию 3000.

Попыток запроса

Количество попыток запроса. По умолчанию 2.

Период проверки входящих

Период проверки входящих данных от считывателя в миллисекундах. Если -1, то входящие проверяются только по событию порта. По умолчанию -1.

Период подключения

Период между попытками подключиться к считывателю в миллисекундах. Если -1, то автоматически не переподключается. По умолчанию 3000.

Виганд

Если включено, то указывает компоненте, что считыватель подключен к Z-2 Base по Wiegand, иначе – по Dallas. По умолчанию выключено.

Бинарный режим

Если включено, то используется бинарный режим, если режим поддерживается считывателем, иначе – ASCII-режим. Поддерживается Z-2 USB MF, Z-2 USB MF-I, Matrix III Net, RF-1996. По умолчанию включено.

Разные коды кнопок брелка IL-100

Если включено, то передаёт разные коды кнопок брелка IL-100 независимо от прошивки считывателя (Z-2 EHR, Matrix V), иначе – одинаковые.

Формат	Кнопка 1		Кнопка 2	
	Em+код	16-ричный	Em+код	16-ричный
Одинаковые	Cod433 Fix[FD] 083,64879	0000FD53FD6F	Cod433 Fix[FB] 083,64879	0000FB53FD6F
Разные	Cod433 Fix[F5] 211,64879	0000F5D3FD6F	Cod433 Fix[F5] 179,64879	0000F5B3FD6F

По умолчанию включено.

Использовать NVT

Если включено, то по TCP данные передаются по протоколу NVT (англ. Network Virtual Terminal, [RFC2217](https://www.rfc-editor.org/rfc/rfc2217)), иначе – как простой текст. Используется для считывателей, подключенных к IP конвертеру Z-397. По умолчанию включено.

Формат ключа

Формат, назначаемый считывателю при его подключении. Если выбран "Авто", то при поднесении ключа формат определяется автоматически с помощью фильтров ключей, иначе – используется указанный формат. По умолчанию "Авто".

Группа "Лог"**Уровень лога**

Если включено, то компонент создает лог файл, в который записывает события компоненты. Путь к лог-файлу "%APPDATA%\IronLogic\ZR1C\zr1c.log". По умолчанию "Отключено"

Уровень лога	Описание
Отключено	Лог выключен.
Критичная ошибка	Неожиданные ошибки, которых быть не должно.
Ошибка	Ошибки. Показывает ошибки системных функций, исключения.
Предупреждение	Предупреждения. Показывает возможные проблемы, которые не являются ошибками.
Информация	Уведомления. Показывает полезную информацию, в основном успехи.
Отладка	Отладочные сообщения. Показывает шаги программы, получаемые и отправляемые данные.
Подробно	Подробные отладочные сообщения. Показывает каждую мелочь.

Макс.размер лог файла (Мбайт)

Максимальный размер лог файла (МБайт). Если = -1, то не ограничено. Если размер файла превысил максимум, то лог файл копируется (с заменой) в "ZR1C.log.old", и затем лог файл очищается. По умолчанию -1;

Контекстное меню поля "Уровень лога" предлагает следующие команды:

- **Открыть лог** – открывает лог файл в текстовом редакторе. Эта команда доступна, когда файл существует;
- **Перейти к лог файлу** – открывает Проводник и выделяет лог файл;
- **Очистить лог** – удаляет содержимое лог файла.

Группа "Доп.внешние события"

Включено

Если включено, то вызывает дополнительные внешние события, которые выбраны ниже. По умолчанию выключено.

УстройствоПодключено

Разрешает вызов внешнего события "[УстройствоПодключено](#)". Вызывается при успешном подключении к считывателю: после вызова метода [Подключить](#) и при автоматическом восстановлении связи. По умолчанию включено.

УстройствоОтключено

Разрешает вызов внешнего события "[УстройствоОтключено](#)". Вызывается при отключении от считывателя: при вызове метода [Отключить](#) и при потере связи. По умолчанию включено.

НетКарты

Разрешает вызов внешнего события "[НетКарты](#)". Вызывается при удалении карты из поля считывателя. По умолчанию включено.

Команда

Разрешает вызов внешнего события "[Команда](#)". Вызывается после выполнения асинхронного метода (см. метод [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#)). По умолчанию включено.

Группа "Настройки компоненты ZR1C"

Хранилище

Место, в которое сохраняются настройки. По умолчанию "Реестр HKCU (для текущего пользователя)".

Значение	Описание
Реестр HKCU (для текущего пользователя)	Ветка реестра HKEY_CURRENT_USER, предназначенная для хранения настроек текущего пользователя. Полный путь "HKCU\Software\IronLogic\ZR1C".
Реестр HKLM (для всех пользователей)	Ветка реестра HKEY_LOCAL_MACHINE, предназначенная для хранения настроек компьютера. По умолчанию для записи в эту ветку требуются права администратора. Полный путь "HKLM\Software\IronLogic\ZR1C".

Загрузить при старте ZR1C

Если включено, то при загрузке компоненты автоматически загружаются настройки из реестра. По умолчанию включено.

Сохранить при завершении ZR1C

Если включено, то перед выгрузкой компоненты автоматически сохраняются изменения настроек в реестр. По умолчанию выключено.

Контекстное меню флагов предлагает следующие команды:

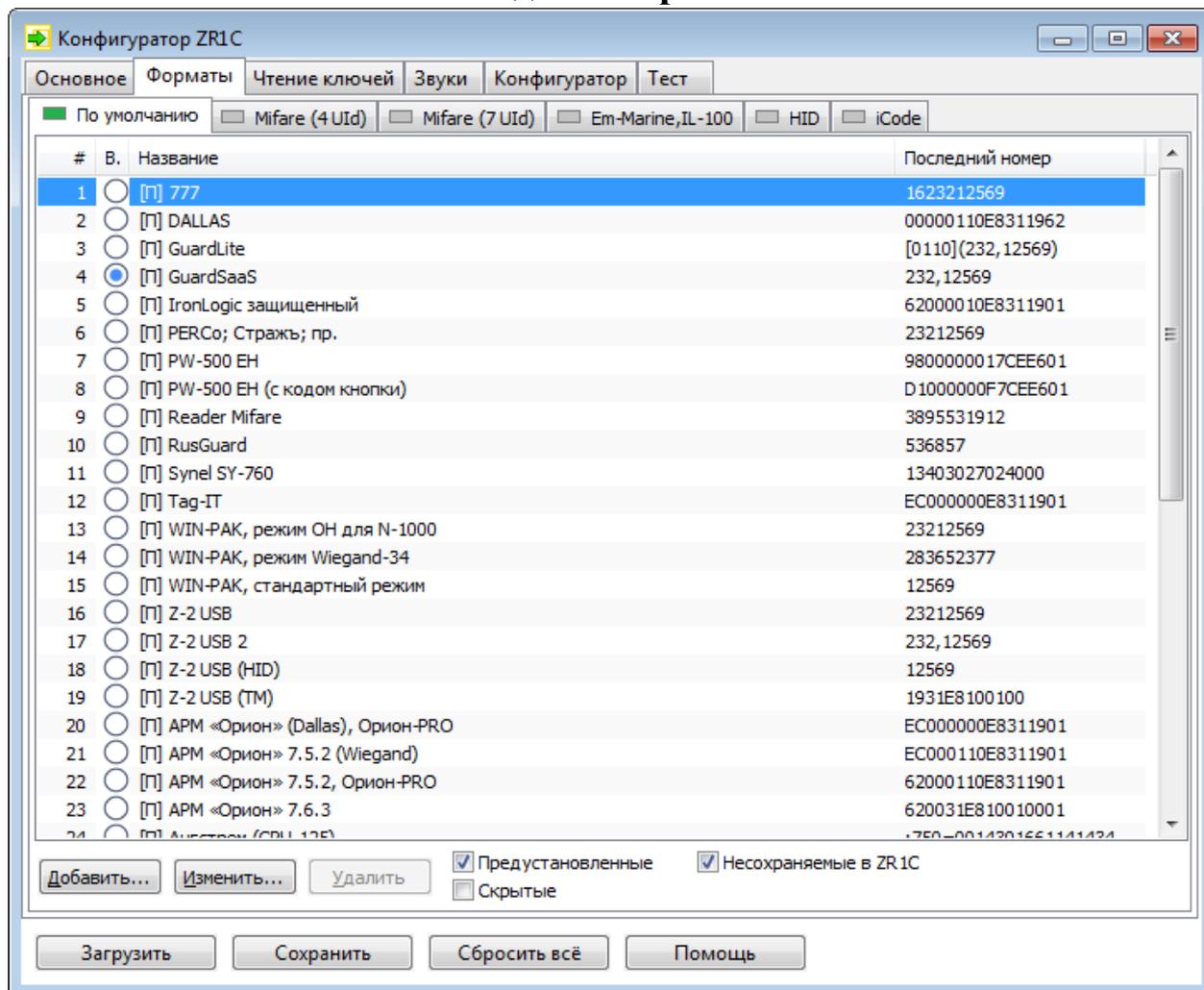
- **Открыть реестр** – запускает стандартную программу "Редактор реестра" (regedit.exe);
- **Перейти к настройкам в реестре** – открывает в Редакторе реестра ветку настроек компоненты. Команда доступна, когда установлена программа "regjump.exe";
- **Удалить настройки в реестре** – удаляет в реестре ветку настроек компоненты;

Кнопки **"Загрузить"** и **"Сохранить"** – загружает/сохраняет настройки компоненты (параметры подключения к считывателю и список включенных форматов) из реестра и настройки конфигуратора (список форматов и список групп типов ключей).

Кнопка **"Сбросить всё"** – сбрасывает все настройки на значения по умолчанию.

Кнопка **"Помощь"** – показывает этот файл справки.

Вкладка "Форматы"



Эта страница предназначена для настройки форматов номеров ключей, передаваемых от считывателя в IC.

В таблице отображается список форматов, которые определяют в каком виде номер ключа должен передаваться в IC.

Кнопка **"Добавить..."** позволяет создать новый формат, при нажатии на неё появляется окно [«Параметры формата»](#).

Кнопка **"Изменить..."** позволяет изменить один или несколько форматов, выделенных в списке, при нажатии на неё появляется окно [«Параметры формата»](#).

Кнопка **"Удалить"** позволяет удалить 1 или несколько форматов, выделенных в списке, при нажатии на неё если флаг подтверждения [«Удаление элемента списка»](#) установлен, то запрашивает подтверждение удаления.

Кнопки **"Загрузить"** и **"Сохранить"** – загружает/сохраняет настройки компоненты (параметры подключения к считывателю и список включенных форматов) из реестра и настройки конфигуратора (список форматов и список групп типов ключей).

Форматы с установленным флагом "Всегда сохранять в настройках ZR1C" сохраняются в настройки компоненты (при нажатии кнопки "Сохранить"), иначе – сохраняются в настройках конфигуратора.

Настройка таблицы

Форматы в таблице можно сортировать кликом ЛКМ по столбцу.

Порядок столбцов можно менять, перетаскивая столбец ЛКМ.

Столбцы таблицы можно настроить кликнув ПКМ по столбцу и выпадающем меню выбрать [«Настроить столбцы...»](#).

Столбец "Последний номер" отображает форматированный номер последнего ключа, полученного от считывателя или введённого в окне [«Параметры ключа»](#) (вызывается командой "Изменить последний ключ..." в контекстном меню списка форматов).

Флаг "**Предустановленные**" – если установлен, то в таблице отображаются предустановленные форматы (с префиксом "[П]"), иначе – скрываются.

Флаг "**Скрытые**" – если установлен, то в таблице отображаются форматы, не предназначенные для текущей вкладки фильтра ключей (предназначенные выбираются в окне [«Параметры ключа»](#)).

Флаг "**Несохраняемые в ZR1C**" – если установлен, то в таблице отображаются невыбранные форматы со снятым флагом "Всегда сохранять в настройках ZR1C", иначе – такие форматы скрываются.

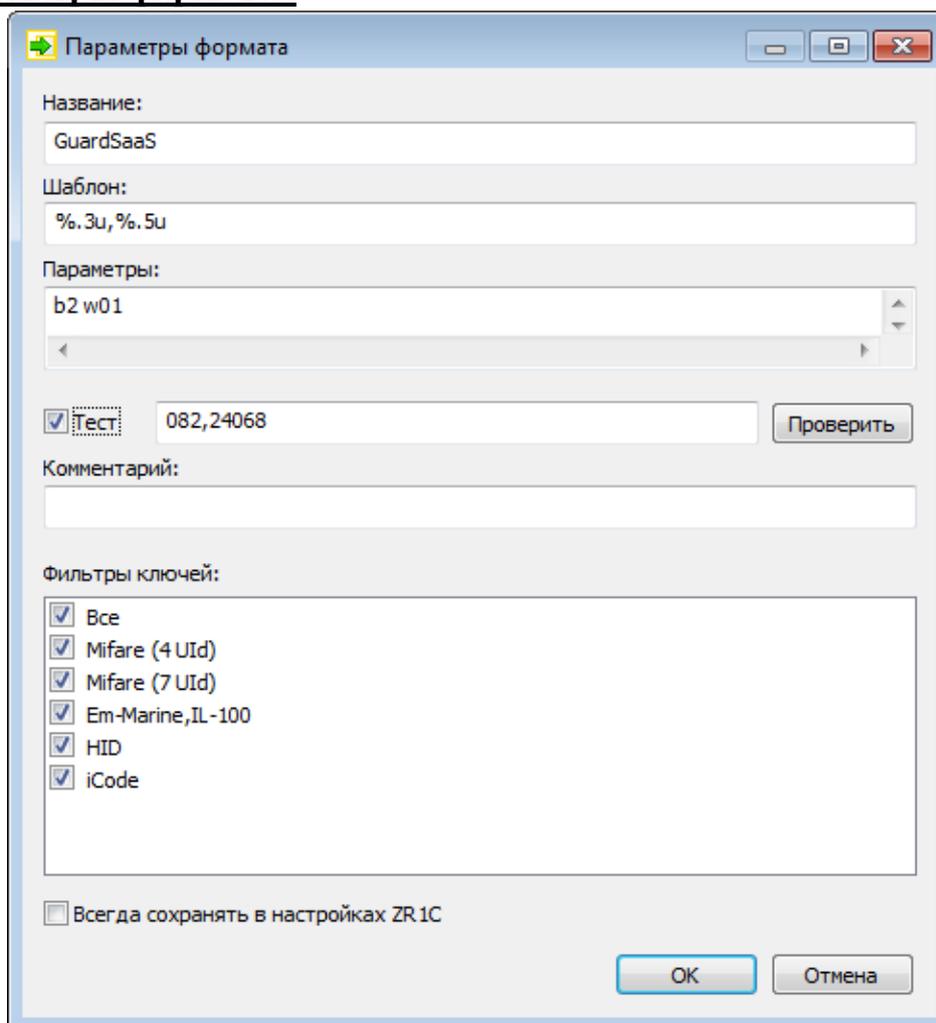
Чтобы импортировать или экспортировать в файл список форматов нужно кликнуть ПКМ по списку и выпавшем меню выбрать подменю "**Импорт**" / "**Экспорт**", затем нужный формат файла (CSV, Microsoft Excel, OpenOffice/LibreOffice Calc).

Фильтры ключей

Фильтры ключей предназначены для определения формата для ключа (по типу ключа и длине номера ключа). Каждому фильтру можно назначить один формат. При считывании ключа компонента ищет подходящий фильтр (исключая фильтра "Все"), если не нашла, то использует фильтр "Все", затем использует формат, назначенный фильтру, для преобразования ключа в текстовую строку, которую затем передаёт во внешнее событие 1С.

Фильтры настраиваются с помощью команды [«Настроить вкладки...»](#) в контекстном меню вкладок;

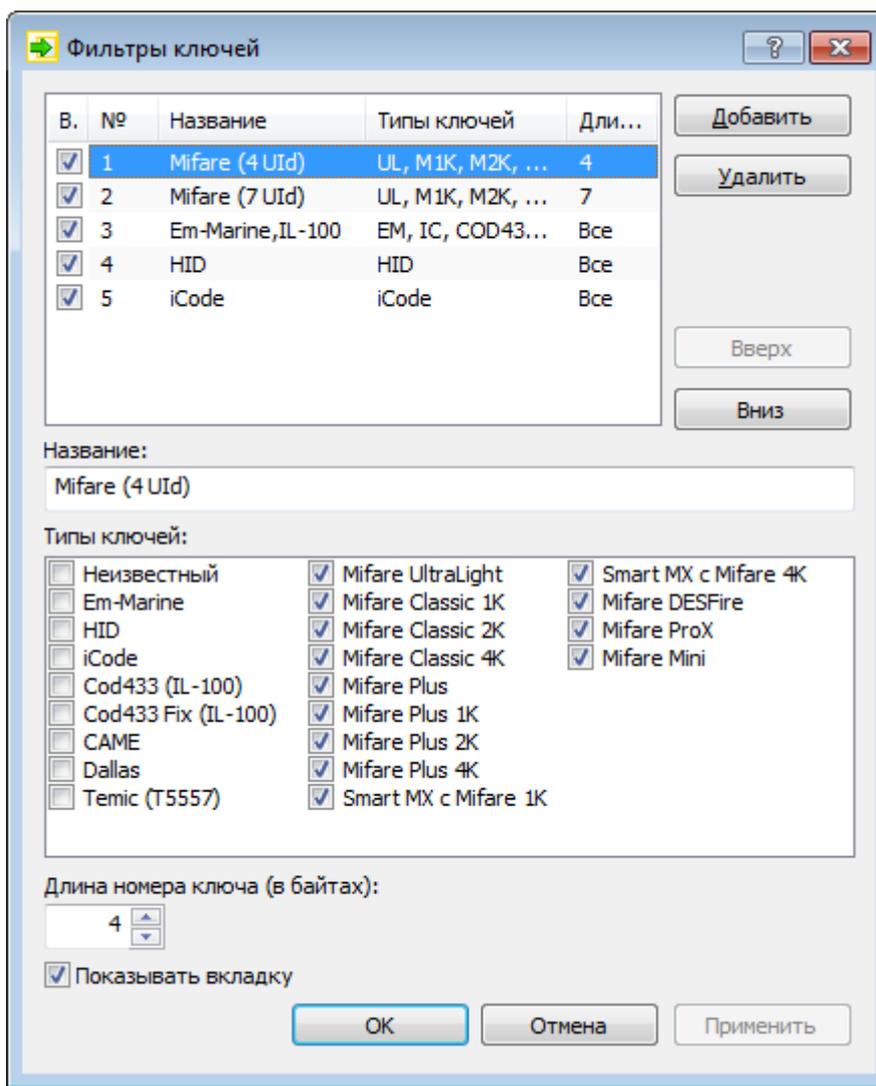
Второй столбец "Выбор" позволяет выбрать один формат, который будет использоваться для текущего фильтра ключей.

Окно "Параметры формата"


Эта окно предназначено для создания нового или изменения существующего формата.

Параметр	Описание
Название	Название формата. Должно быть не пустым и уникальным. Обязательное поле.
Шаблон	Текст шаблона , используемого для форматирования номера ключа. Обязательное поле.
Параметры	Текст параметров , используемых для форматирования номера ключа. Количество параметров должно быть не меньше, чем количество спецификаций (%) в шаблоне.
Флаг "Тест"	Если установлен, то в текстовом поле справа отображается пример форматирования ключа (обновляется при поднесении ключа к считывателю).
Комментарий	Произвольный комментарий.
Фильтры ключей	Список фильтров ключей, для которых предназначен этот формат. Если флаг установлен, то этот формат отображается на вкладке фильтра , иначе – скрыт. Список фильтров настраивается в окне « Фильтры ключей ».
Всегда сохранять в настройках ZR1C	Если установлен, то формат сохраняется в настройках компоненты даже если не выбран, иначе – только в настройках конфигуратора.

Кнопка "**Проверить**" проверяет корректность значений полей "Шаблон" и "Параметры".

Окно "Фильтры ключей"

Это окно предназначено для настройки фильтров ключей, которые отображаются в виде вкладок на вкладке «[Форматы](#)». Чтобы вызвать окно нужно на вкладке «[Форматы](#)» кликнуть ПКМ по вкладкам фильтров ("По умолчанию", "Mifare UL,DF,DS", ...) и в выпадающем меню выбрать "Настроить вкладки...".

1. Список фильтров

Показывает существующие фильтры.

Список фильтров отображаются в виде вкладок на вкладке «[Форматы](#)».

2. Кнопки редактирования списка

Кнопка "Добавить" – добавляет в список новый фильтр. Максимум можно добавить 31 фильтр.

Кнопка "Удалить" – удаляет фильтр, выделенный в списке.

Кнопка "Вверх" – перемещает фильтр, выделенный в списке, на 1 строку выше.

Кнопка "Вниз" – перемещает фильтр, выделенный в списке, на 1 строку ниже.

3. Параметры фильтра

Позволяет редактировать параметры фильтров, выделенных в списке.

Поле "Название" – произвольное название фильтра. Поле должно быть не пустым.

Поле "Типы ключей" – список разрешенных типов ключей.

Поле "Длина номера ключа (в байтах)" – разрешенный размер номера ключа. Если равен 0, то размер не проверяется.

Флаг "Показывать вкладку" – если установлен, то вкладка фильтра отображается на вкладке «[Форматы](#)», иначе – не отображается (скрыта).

4. Кнопка "ОК"

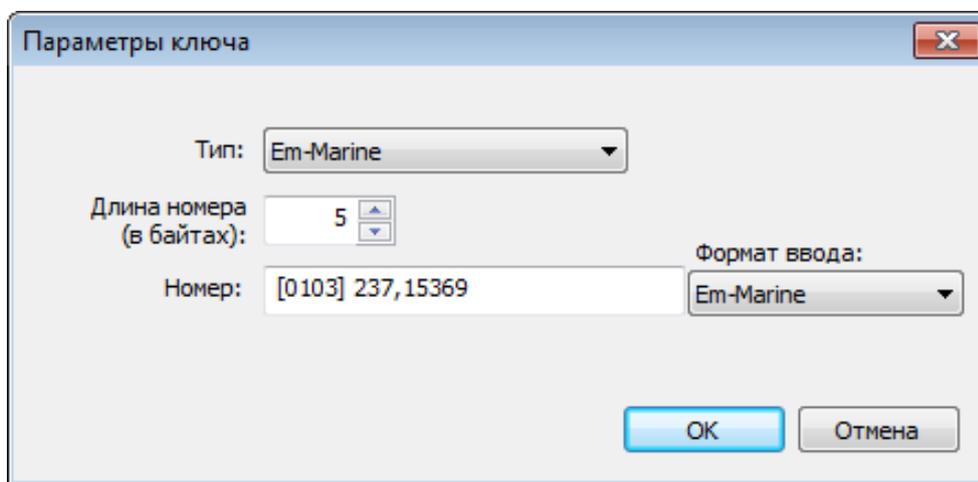
Сохраняет изменения и закрывает окно.

5. Кнопка "Отмена"

Закрывает окно без сохранения изменений.

6. Кнопка "Применить"

Сохраняет изменения (окно не закрывает).

Окно "Параметры ключа"


Это окно предназначено для ручного ввода номера ключа для форматирования и отображения на вкладке «[Форматы](#)» в столбце "Последний номер". Чтобы вызвать окно нужно на вкладке «[Форматы](#)» кликнуть ПКМ по списку форматов и в выпадающем меню выбрать "**Изменить последний ключ...**".

1. Тип

Тип ключа.

2. Длина номера

Размер номера ключа в байтах от 1 до 16.

3. Номер

Номер ключа. Номер можно вводить в одном из форматов:

- **Em-Marine** – 2 числа в 10-тичном виде (третий байт и младшие 2 байта), разделенные запятой (','), например, "235,15369". Если длина номера > 3, то в квадратных скобках – код производителя в 16-ричном виде, например, "[0103] 235,15369";
- **IID** – 1 число в 16-ричном виде в квадратных скобках (код производителя) и 1 число в 10-тичном виде (номер, младшие 2 байта), например, "[0103EB] 15369".
- **16-ричное** – 1 число в шестнадцатеричном виде (младший байт справа), например, "0103EB3C09";
- **10-тичное + код** – 1 число в десятичном виде, например, "4360715273". Если длина номера > 8, то в квадратных скобках – старшие байты в 16-ричном виде, например, "[000000] 4360715273";

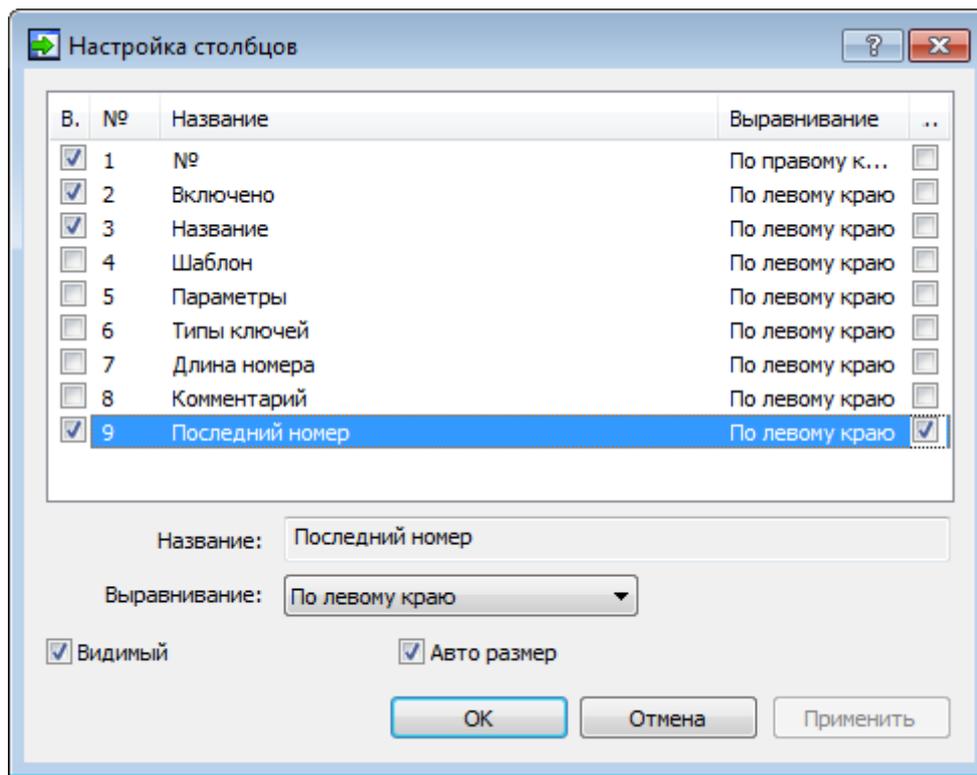
 Примечание

Формат ввода не влияет на то, как ключ будет форматироваться при передаче.

4. Формат ввода – позволяет изменить формат номера, вводимого в поле "Номер".

5. Кнопка "ОК" – сохраняет изменения и закрывает окно.

6. Кнопка "Отмена" – закрывает окно "Параметры ключа" без сохранения изменений.

Окно «Настройка столбцов»

Это окно предназначено для настройки столбцов таблицы. Это окно вызывается кликом ПКМ по заголовку таблицы > в контекстном меню выбрать команду "Настроить столбцы...".

1. Список столбцов

Показывает названия всех существующих столбцов таблицы.

Список столбцов (их названия) отображаются в заголовке таблицы (в верхней части).

2. Выделенный столбец

Выделенный элемент в списке столбцов, параметры которого можно редактировать на панели "Параметры столбца".

3. Параметры столбца

Позволяет редактировать параметры столбцов, выделенных в списке.

Поле "**Название**" – название столбца, которое отображаться в заголовке таблицы. Поле не доступно для редактирования.

Поле "**Выравнивание**" – определяет положение текста в столбце.

Флаг "**Видимый**" – если установлен, то столбец отображается в таблице, иначе – скрыт.

Флаг "**Авто размер**" – если установлен, то ширина столбца занимает всё свободное пространство, иначе – ширина задаётся пользователем.

4. Кнопка "ОК"

Сохраняет изменения и закрывает окно.

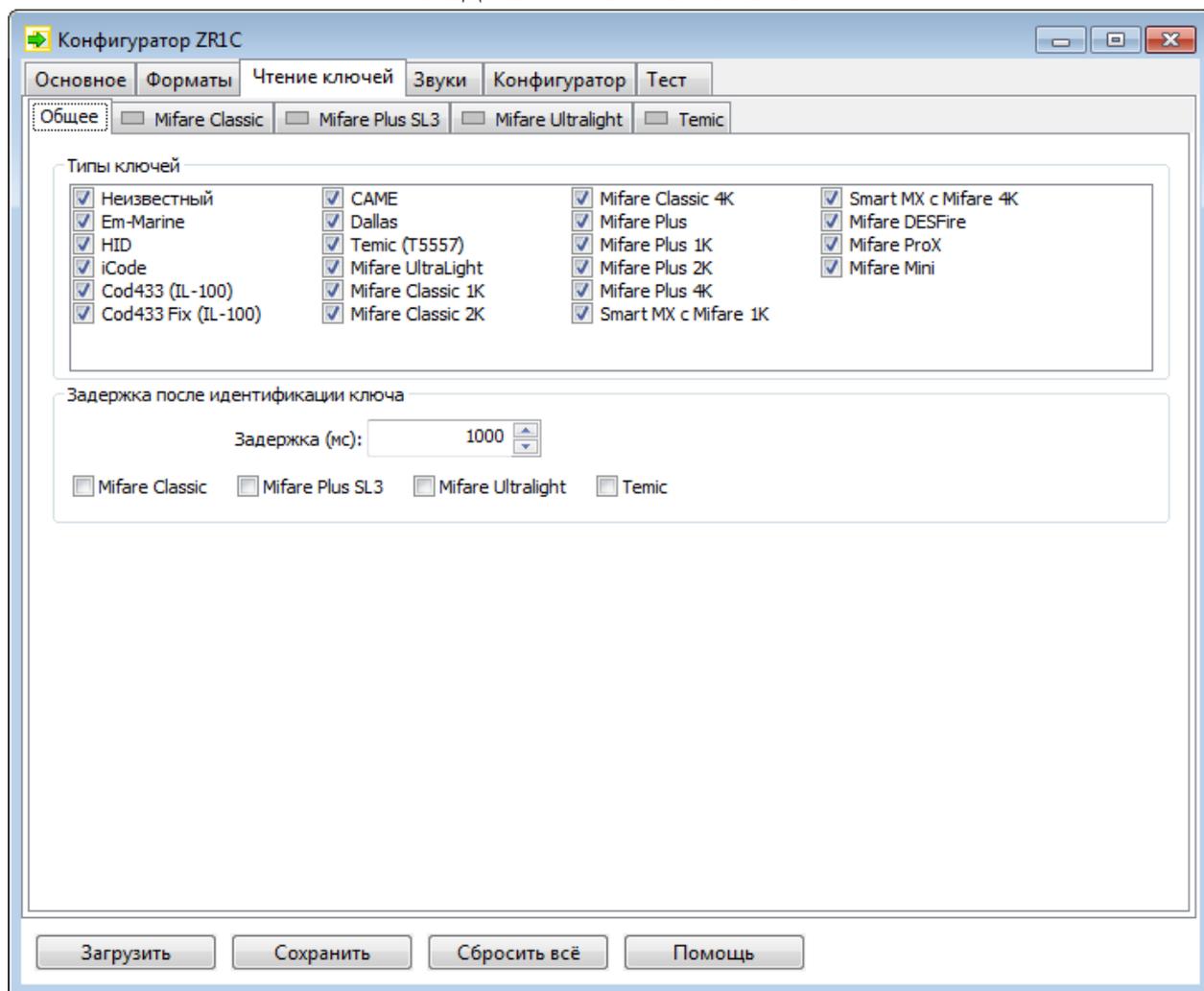
5. Кнопка "Отмена"

Закрывает окно без сохранения изменений.

6. Кнопка "Применить"

Сохраняет изменения (окно не закрывает).

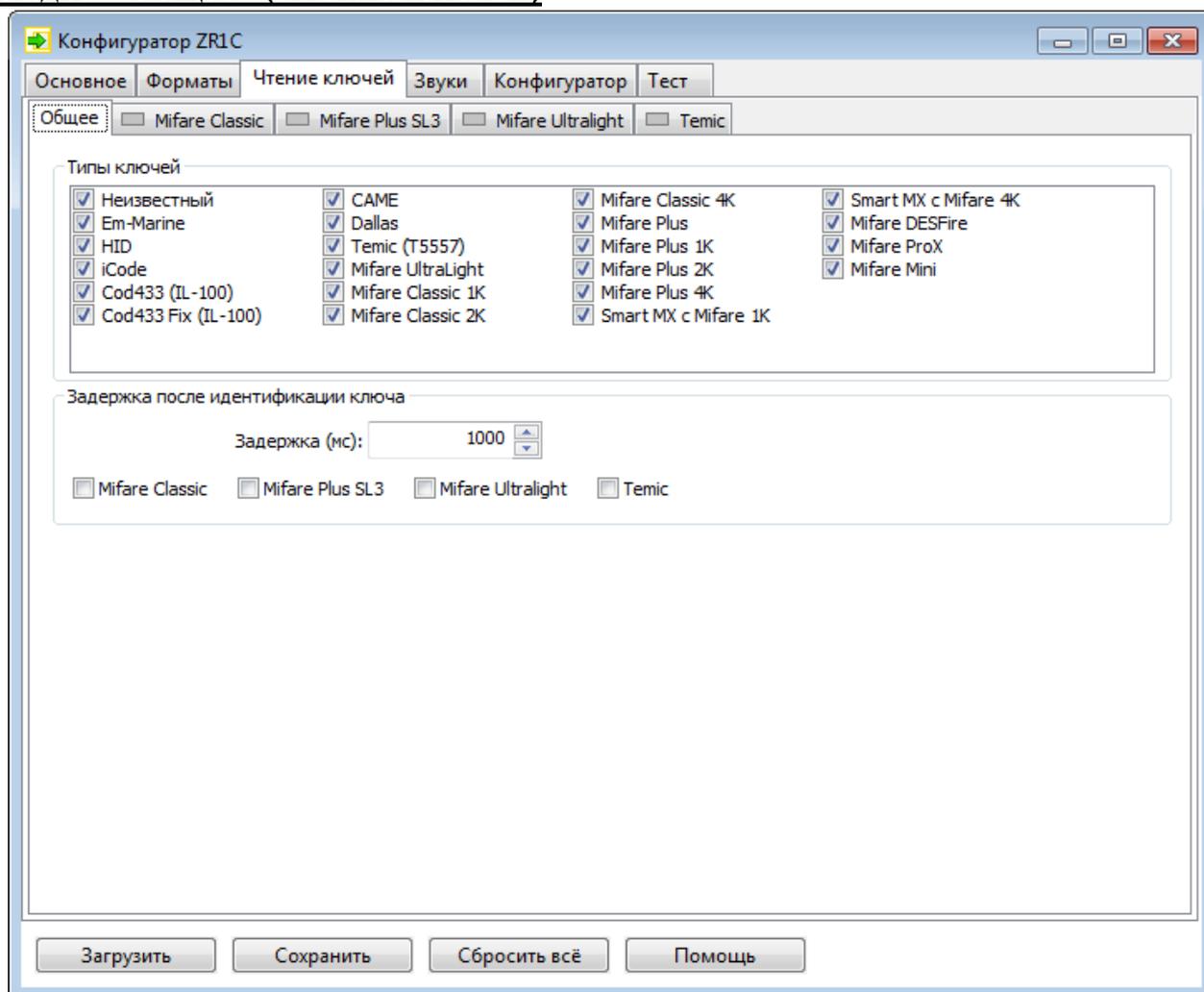
Вкладка "Чтение ключей"



Эта страница предназначена для настройки чтения ключей.

Вкладка «[Общее](#)» предназначена для настройки фильтра ключей и задержки после идентификации ключа.

Вкладки «[Mifare Classic](#)», «[Mifare Plus SL3](#)», «[Mifare Ultralight](#)», «[Temic](#)» предназначены для настройки чтения номера из памяти ключа для передачи в IC.

Вкладка "Общее" (Чтение ключей)

Эта страница предназначена для настройки общих параметров чтения ключей.

Группа "Типы ключей"

Позволяет выбрать типы ключей, разрешенных для передачи.

Группа "Задержка после идентификации ключа"

Позволяет приостановить на время цикл сканирования ключей после обнаружения нового ключа заданного типа, чтобы ускорить работу методов чтения/записи памяти ключей:

- [ЧитатьMifareC \(ReadMifareC\)](#)
- [ПисатьMifareC \(WriteMifareC\)](#)
- [ЧитатьMifareP \(ReadMifareP\)](#)
- [ПисатьMifareP \(WriteMifareP\)](#)
- [ЧитатьMifareUL \(ReadMifareUL\)](#)
- [ПисатьMifareUL \(WriteMifareUL\)](#)
- [ЧитатьTemic \(ReadTemic\)](#)
- [ПисатьTemic \(WriteTemic\)](#)

Задержка (мс)

Время в миллисекундах, на которое приостанавливается сканирование после обнаружения ключа. Если = -1, то приостанавливается на неограниченное время пока не будет вызван метод [ПродолжитьСканирование \(ContinueScanning\)](#). По умолчанию 1000;

Mifare Classic

Если включено, то при обнаружении ключа типа [Mifare Classic](#) сканирование приостанавливается. По умолчанию выключено;

Mifare Plus SL3

Если включено, то при обнаружении ключа типа [Mifare Plus](#) с уровнем безопасности SL3 сканирование приостанавливается. По умолчанию выключено;

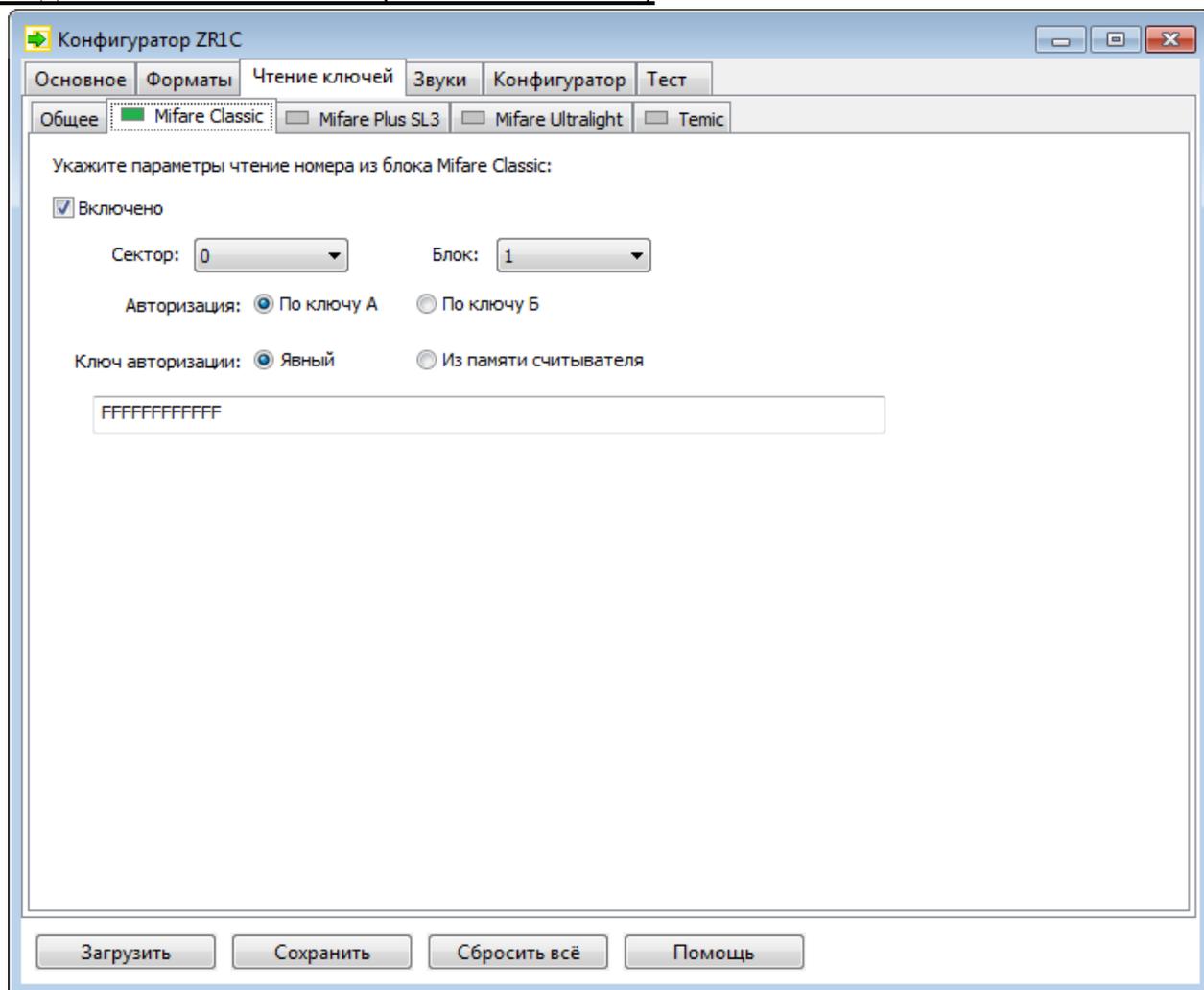
Mifare Ultralight

Если включено, то при обнаружении ключа типа [Mifare Ultralight](#) сканирование приостанавливается. По умолчанию выключено;

Temic

Если включено, то при обнаружении ключа типа [Temic](#) сканирование приостанавливается. По умолчанию выключено.

Вкладка "Mifare Classic" (Чтение ключей)



Эта страница предназначена для настройки чтения номера из памяти Mifare Classic для передачи в 1С.

Включено

Если флаг установлен, то номер ключа считывается из блока Mifare.

Сектор

Номер сектора карты Mifare (1К: от 0..15; 4К: от 0 до 39, см. структуру памяти [Mifare 1К](#) и [Mifare 4К](#)).

Блок

Номер блока в секторе Mifare.

Авторизация

Тип авторизации: по ключу авторизации А или по ключу Б.

Ключ авторизации

Параметры доступа к данным карты Mifare Classic:

Тип пароля	Описание
Явный	Ключ авторизации 6 байт – число в 16-ричном формате (младший байт справа).
Из памяти считывателя	Список ключей авторизации в памяти считывателя.

Вкладка "Mifare Plus SL3" (Чтение ключей)

The screenshot shows the 'Конфигуратор ZR1C' window with the 'Чтение ключей' tab selected. The 'Mifare Plus SL3' option is checked. The interface includes the following elements:

- Checkboxes for 'Общее', 'Mifare Classic', 'Mifare Plus SL3', 'Mifare Ultralight', and 'Temic'.
- Text: 'Укажите параметры чтение номера из блока Mifare Plus SL3:'
- Checkbox: Включено
- Dropdowns: Сектор: 0, Блок: 1
- Radio buttons for 'Авторизация': По ключу А, По ключу Б
- Radio buttons for 'Ключ авторизации': Явный, Из памяти считывателя
- Text input field containing: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
- Buttons at the bottom: Загрузить, Сохранить, Сбросить всё, Помощь

Эта страница предназначена для настройки чтения номера из памяти [Mifare Plus](#) с уровнем безопасности SL3 для передачи в 1С.

Включено

Если флаг установлен, то номер ключа считывается из блока Mifare.

Сектор

Номер сектора карты Mifare.

Блок

Номер блока в секторе Mifare.

Авторизация

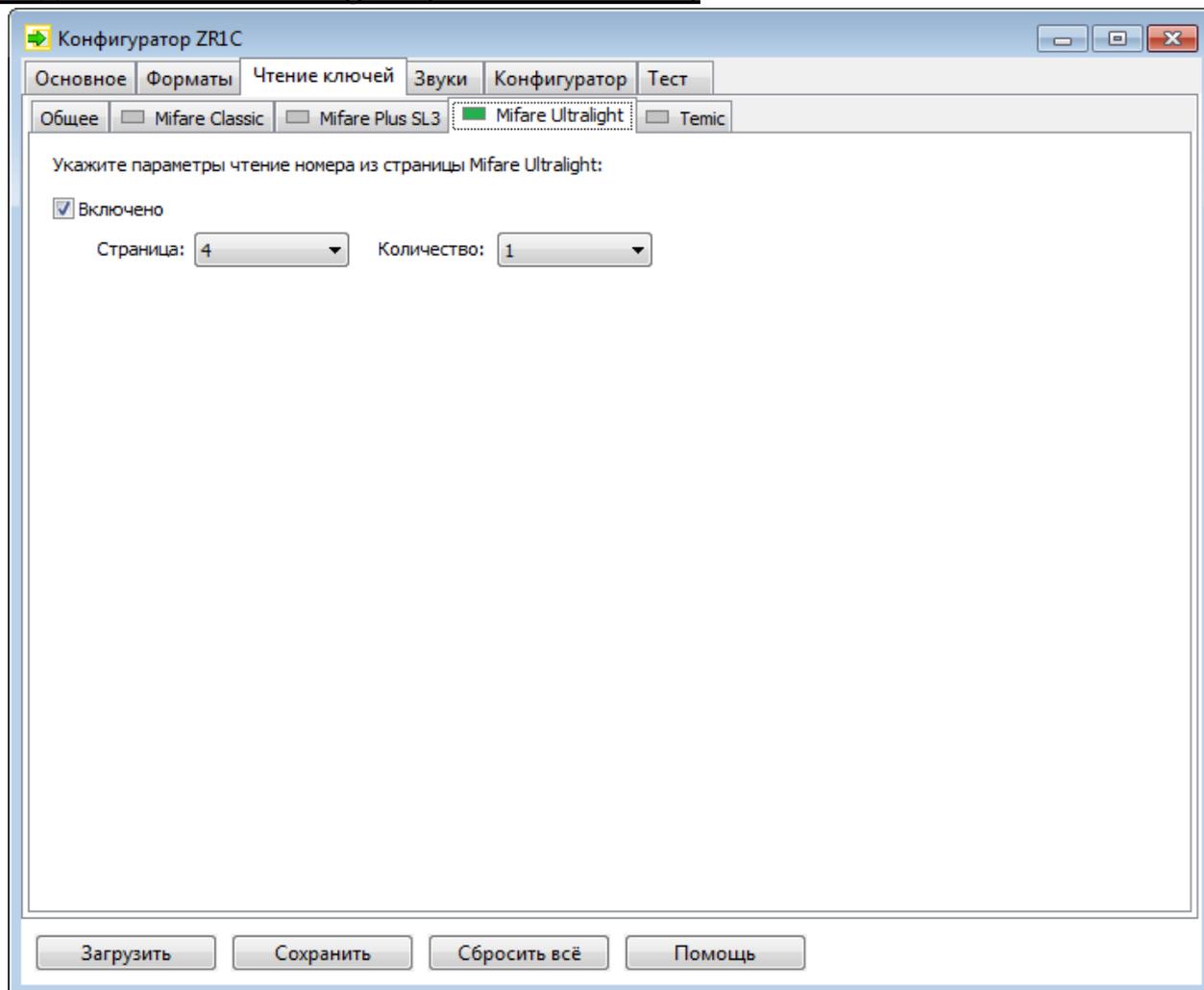
Тип авторизации: по ключу авторизации А или по ключу Б.

Ключ авторизации

Параметры доступа к данным карты Mifare Plus:

Тип пароля	Описание
Явный	Ключ авторизации 16 байт – число в 16-ричном формате (младший байт справа).
Их памяти считывателя	Список ключей авторизации в памяти считывателя.

Вкладка "Mifare Ultralight" (Чтение ключей)



Эта страница предназначена для настройки чтения номера из страницы [Mifare Ultralight](#) для передачи в 1С.

Включено

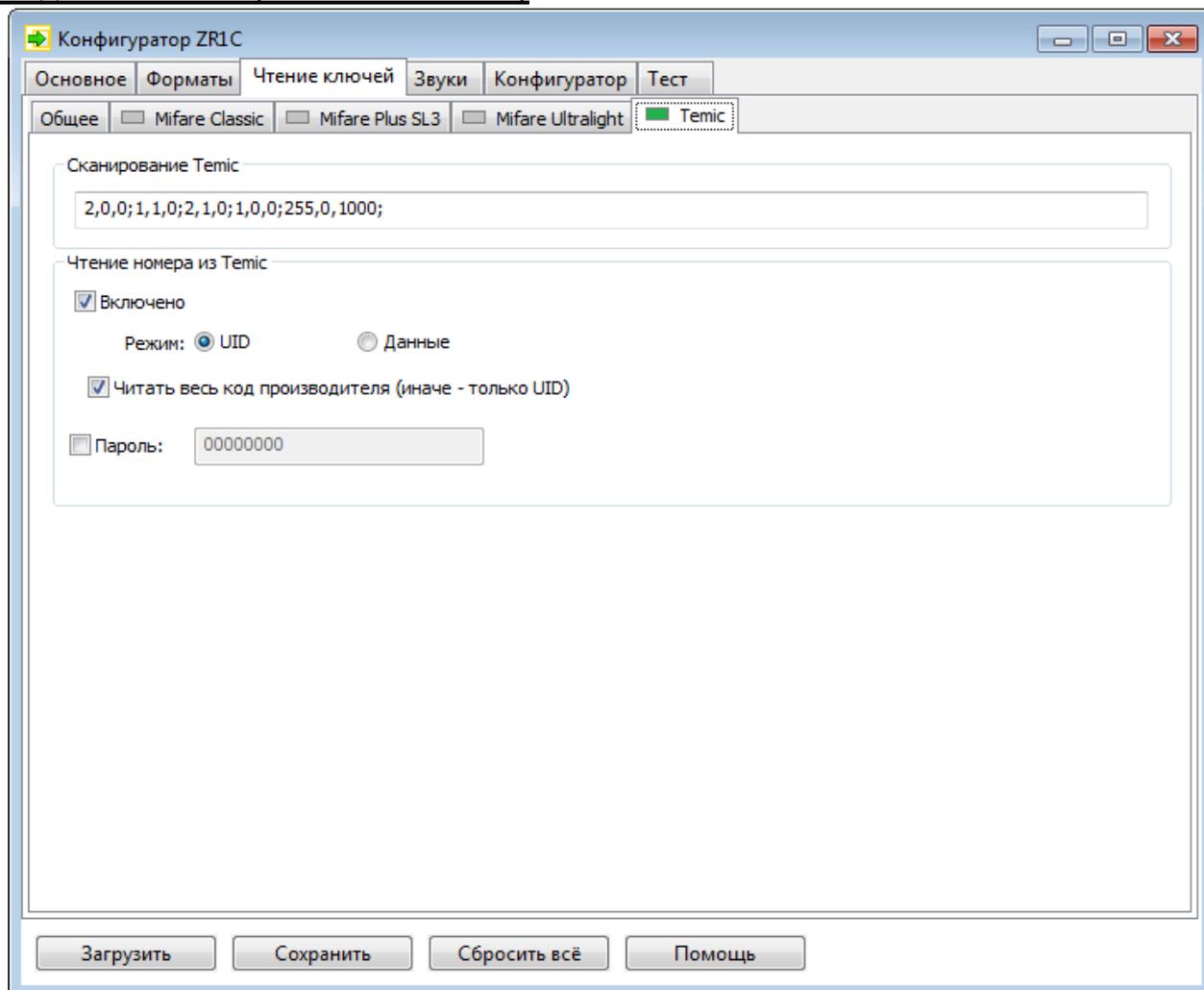
Если флаг установлен, то номер ключа считывается из страницы Mifare Ultralight.

Страница

Номер первой считываемой страницы [Mifare Ultralight](#).

Количество

Количество считываемых страниц Mifare Ultralight.

Вкладка "Temic" (Чтение ключей)

Эта страница предназначена для настройки чтения номера из памяти [Temic](#) для передачи в 1С.

Включено

Разрешает сканирование Temic (иначе – Temic не сканируется). Поддерживается считывателями: [Z-2 \(мод. RD_ALL\)/Z-2 USB](#), [Z-2 \(мод. E HTZ RF\)/Z-2 EHR](#);

Параметры сканирования

Параметры заросов, разделенные символом ';'. Параметры запроса разделены символом '!':

1. Тип – =1 с терминатором, =2 без терминатора. Если =255, то вызывается команда считывателя "reset", которая ненадолго снимает генерацию 125кГц, чтобы считыватель перешел в нормальный режим работы;
2. Скорость – =0 стандарт div64, =1 div32;
3. Задержка – интервал времени в миллисекундах между запросами.

Режим

Определяет откуда читать номер ключа:

- UID – из страницы 1 с данными производителя;
- Данные – из страницы 0 с пользовательскими данными, конфигурацией и паролем.

Читать весь код производителя (иначе – только UID)

Если установлен, читает все 64 бита данных производителя (2 блока), иначе – читает UID из "LotID", "wafer #" и "die on wafer #".

Блок / Бит	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
1	ACL						MFC						ICR						MSN LotID															
2	LotID										wafer #										die on wafer #													

ACL – класс распределения, определенный в ISO/IEC 15963-1 =0xE0;

MFC – Заводской код корпорации Atmel, как определено в ISO/IEC 7816-6 =0x15;

ICR – IC ссылка производителя кремния или тег: верхние 3 бита определяют ревизию IC, нижние 5 битов могут содержать код идентификатора клиента по запросу;

MSN – Заводской серийный номер состоит из: LotID, DPW;

LotID – 5-значный номер партии, например, "38765";

DPW – 20 битов, закодированных как последовательная матрица на номер пластины (с верхними 5 битами =wafer#).

Блок

[Номер блока](#) страницы 0, с которой нужно прочесть номер ключа;

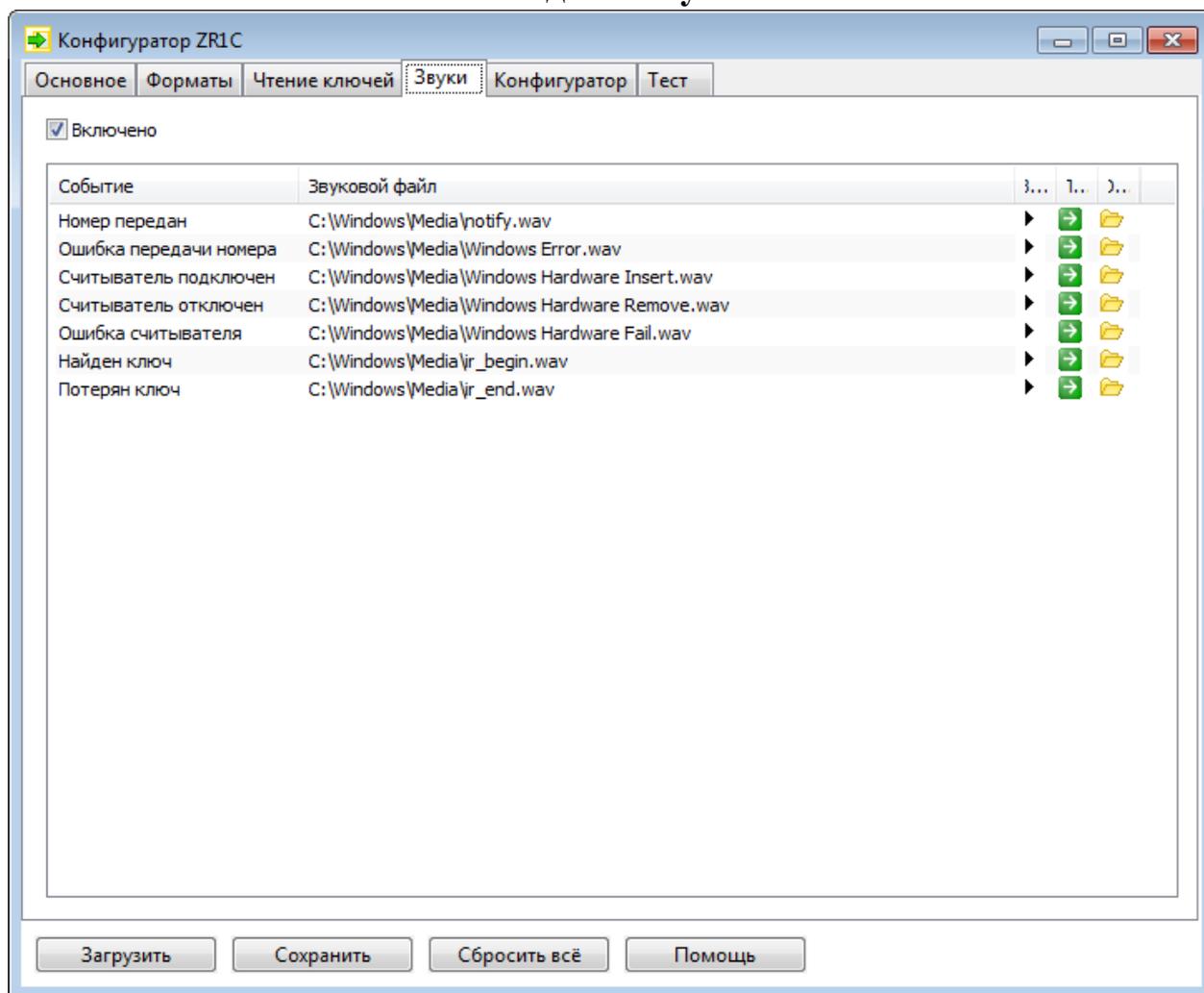
Количество

Количество блоков страницы 0;

Пароль

Пароль для доступа к данным карты Temic, значение – число в 16-ричном формате (4 байта, младший байт справа).

Вкладка "Звуки"



Эта страница предназначена для настройки звуковых уведомлений компоненты ZR1C.

Включено

Если включен, то звуковые уведомления компоненты ZR1C разрешены, иначе – запрещены. По умолчанию выключено;

Таблица

Позволяет выбрать звуковой файл для любого события:

- Номер передан – при успешной передаче номера ключа в IC;

- Ошибка передачи номера – при ошибке передачи номера ключа в 1С;
- Считыватель подключен – при подключении к считывателю;
- Считыватель отключен – при отключении от считывателя;
- Найден ключ – при поднесении карты к считывателю;
- Потерян ключ – при удалении карты от считывателя.

Кнопка  – позволяет выбрать звуковой файл.

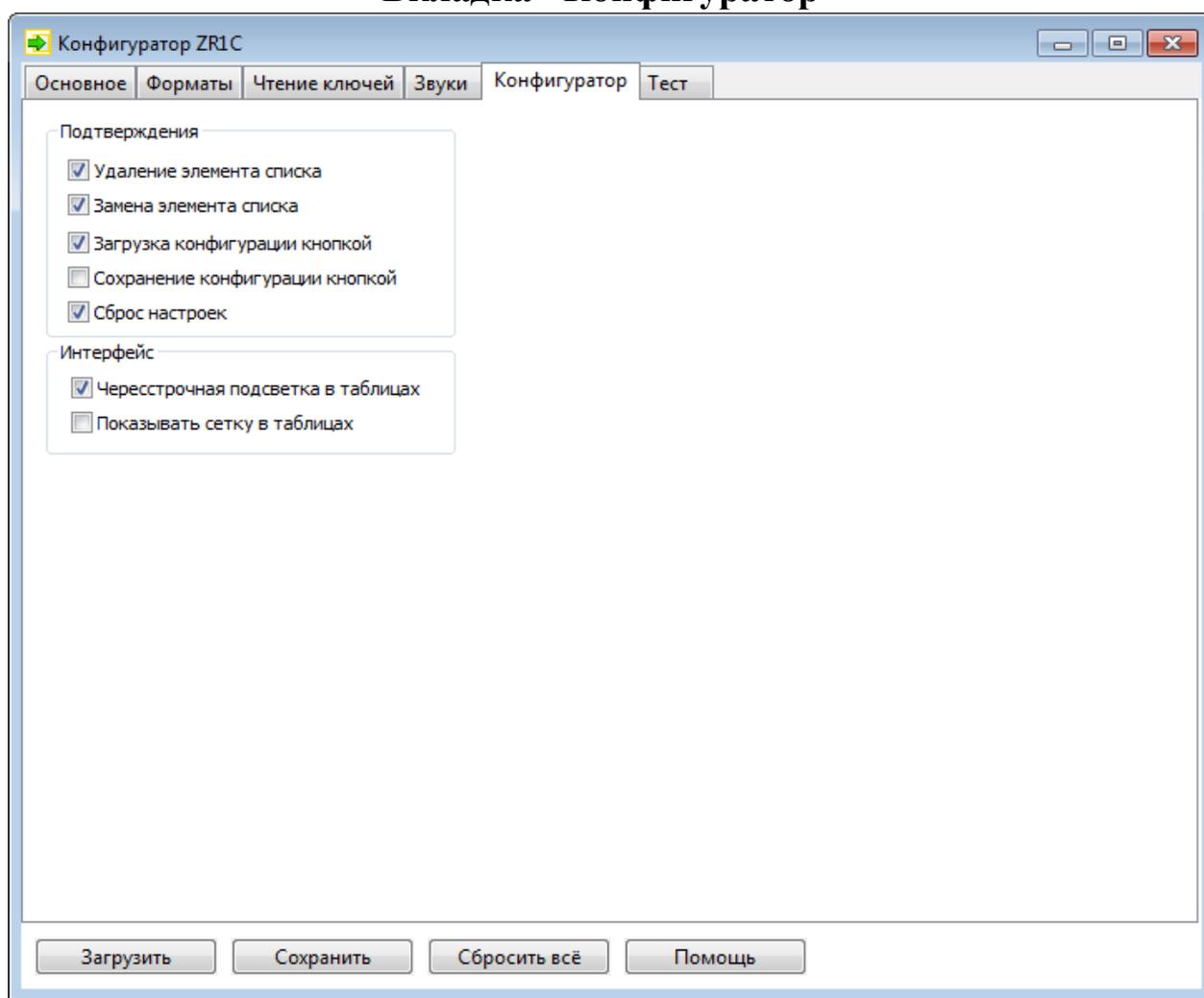
Кнопка  – открывает в проводнике папку, в которой содержится звуковой файл.

Кнопка  – воспроизводит звуковой файл, назначенный событию.

Примечание

Клик ЛКМ по выделенной строке под столбцом "Звуковой файл" открывает выпадающее меню, в котором можно выбрать звуки Windows.

Вкладка "Конфигуратор"



Эта страница предназначена для настройки подтверждений в программе "Конфигуратор ZR1C".

Группа "Подтверждения"

Удаление элемента списка

Если включено, то перед удалением формата и перед удалением фильтра ключей показывает окно с запросом подтверждения. По умолчанию включено;

Замена элемента списка

Если включено, то перед заменой формата показывает окно с запросом подтверждения. По умолчанию включено;

Загрузка конфигурации кнопкой

Если включено, то при нажатии кнопки "Загрузить" запрашивает подтверждение. По умолчанию включено;

Сохранение конфигурации кнопкой

Если включено, то при нажатии кнопки "Сохранить" запрашивает подтверждение. По умолчанию выключено;

Сброс настроек

Если включено, то при нажатии кнопки "Сбросить всё" запрашивает подтверждение. По умолчанию включено.

Группа "Интерфейс"

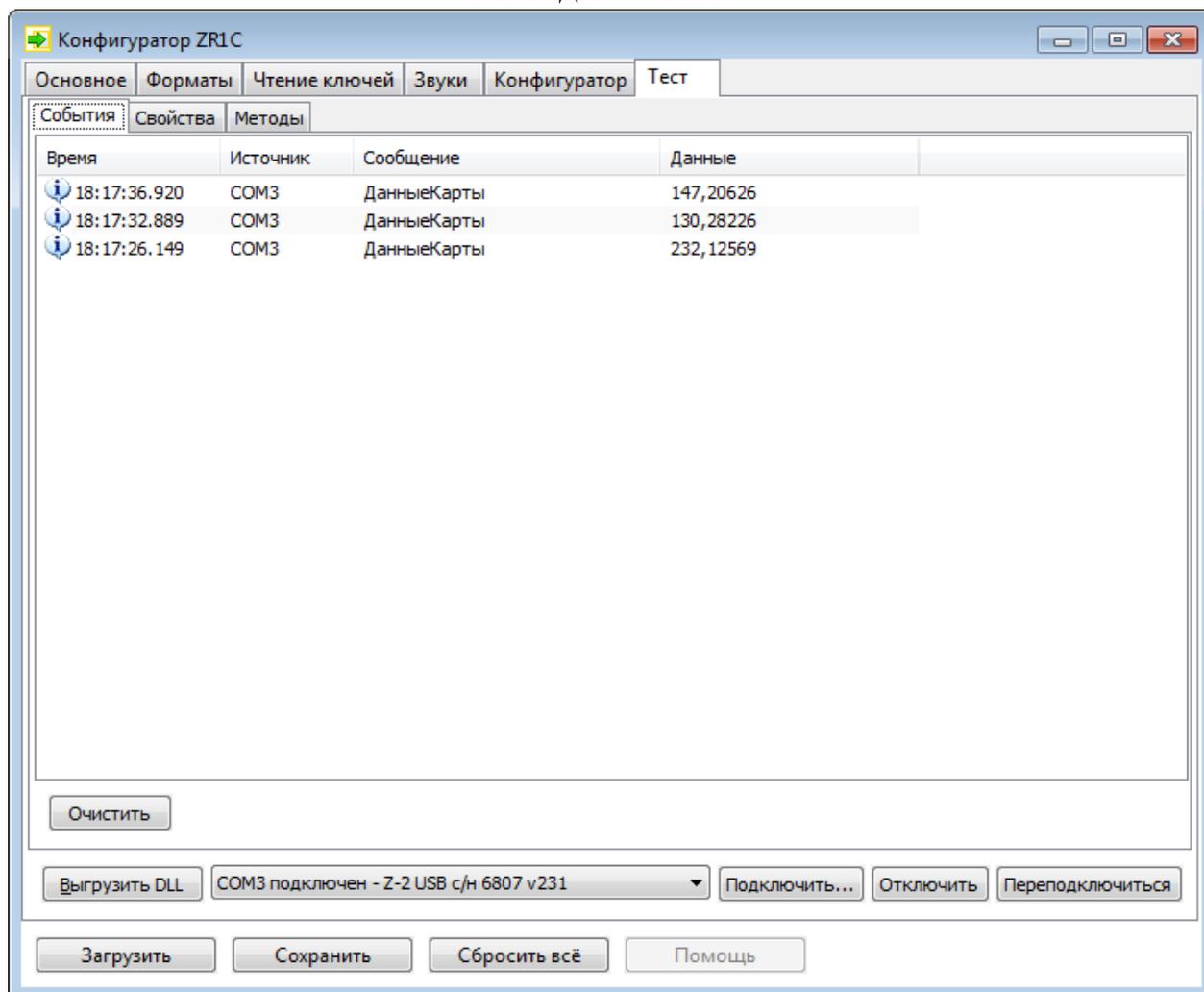
Чересстрочная подсветка в таблицах

Если включено, то рисует нечётные строки таблиц нестандартным цветом. По умолчанию включено;

Показывать сетку в таблицах

Если включено, то рисует в таблицах разделительные линии между столбцами и строками. По умолчанию выключено.

Вкладка "Тест"



Эта страница предназначена для тестирования компоненты без использования IC.
Вкладка "События" – позволяет видеть сообщения о событиях компоненты.

Вкладка "Свойства" – показывает список свойств компоненты.

Вкладка "Методы" – показывает список методов компоненты.

Кнопка "Загрузить DLL "/"Выгрузить DLL " – загружает или выгружает (если уже загружена) библиотеку компоненты "ZR1C<суффикс>.dll" (суффикс копируется из имени программы конфигуратора "ConfigZR1C<суффикс>.exe", например, "ConfigZR1C64.exe" загружает "ZR1C64.dll").

Выпадающий список – отображает список подключенных считывателей (с помощью кнопки "Подключить").

Кнопка "Подключить..." – вызывает окно [«Подключение»](#), позволяющее подключиться к считывателю.

Кнопка "Отключить" – отключается от считывателя, выбранного в выпадающем списке, вызывая метод [Отключить \(Close\)](#).

Кнопка "Переподключиться" – переподключается к считывателю, вызывая метод [Переподключиться \(Reconnect\)](#).

Окно "Подключение"

Эта окно предназначено для подключения к считывателю.

Кнопка **"Записать и закрыть"** подключается к считывателю и закрывает окно.

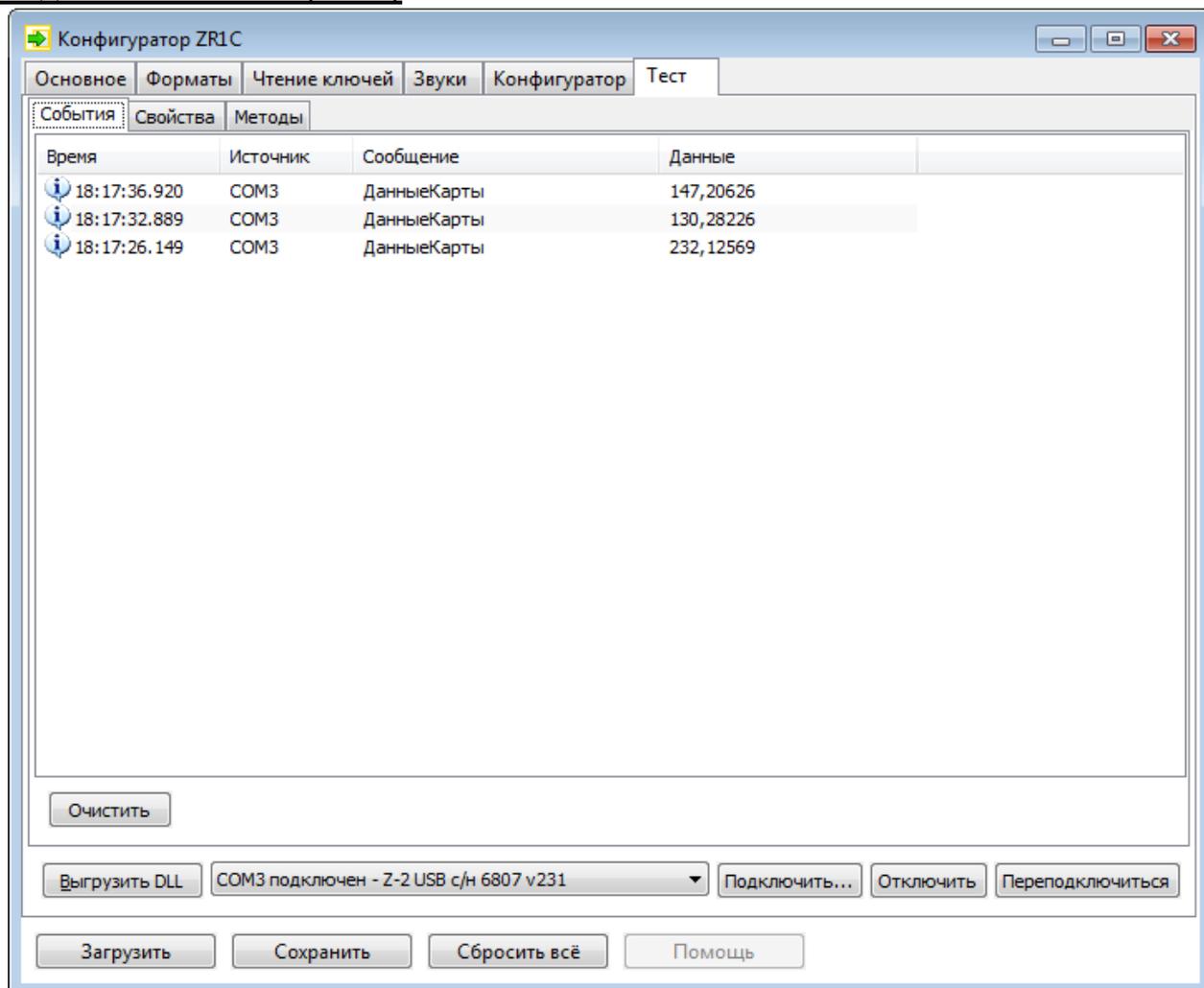
Кнопка **"Функции"** показывает сервисное меню с командой "Тест устройства", которая опрашивает считыватель и показывает информацию о нём.

Поле **"Драйвер"** всегда отображает значение "Установлен".

Поле **"Версия"** отображает версию загруженной компоненты.

Группа **"Параметры подключения"** позволяет настроить параметры подключения к считывателю.

Название параметра	Описание
Параметры подключения	
Модель	Модель считывателя. Если "Auto", то определяется автоматически. Выбор определенной модели (не "Auto") может ускорить подключение к считывателю.
Порт	Имя порта считывателя (com-порт или адрес порта ip-конвертера, к которому подключен считыватель). Если "Auto", то используется первый найденный порт считывателя. Если используется сетевая прошивка Matrix III Rd-All, то имя порта должно быть в формате "COM3\123", где "COM3" – порт конвертера, "123" – серийный номер (с/н) считывателя.
Скорость	Предельная скорость считывателя. Если =0, то выбирается автоматически (максимальная из поддерживаемых считывателем). Позволяет ограничить скорость работы со считывателем.
Виганд	Если включено, то указывает компоненте, что считыватель подключен по Wiegand, иначе – по Dallas. Используется для настройки формата Z-2 Base.
Формат ключа	
Формат	Формат номера ключа. Если = "Авто", то используется формат, соответствующий типу ключа (схема выбора формата), иначе – используется только указанный формат. Форматы можно редактировать 2 способами: 1) в "Конфигураторе ZR1C" на вкладке "Форматы", 2) в ZR1C с помощью методов: ОчиститьФорматы , ВставитьФормат и т.д.
Группа "Параметры устройства"	
Префикс	Не используется компонентой ZR1C, требуется 1С, рекомендуется не изменять значения по умолчанию.
Суффикс	

Вкладка "События" (Тест)

Эта страница предназначена для тестирования событий компоненты без использования IC. В таблице отображаются записи о внешних событиях компоненты (наверху последнее событие). В контекстном меню таблицы доступны команды:

- "Язык" – позволяет выбрать язык компоненты: английский или русский;
- "Очистить" – очищает содержимое таблицы;

События появляются в таблице после подключения к считывателю и подносе карты в поле считывателя.

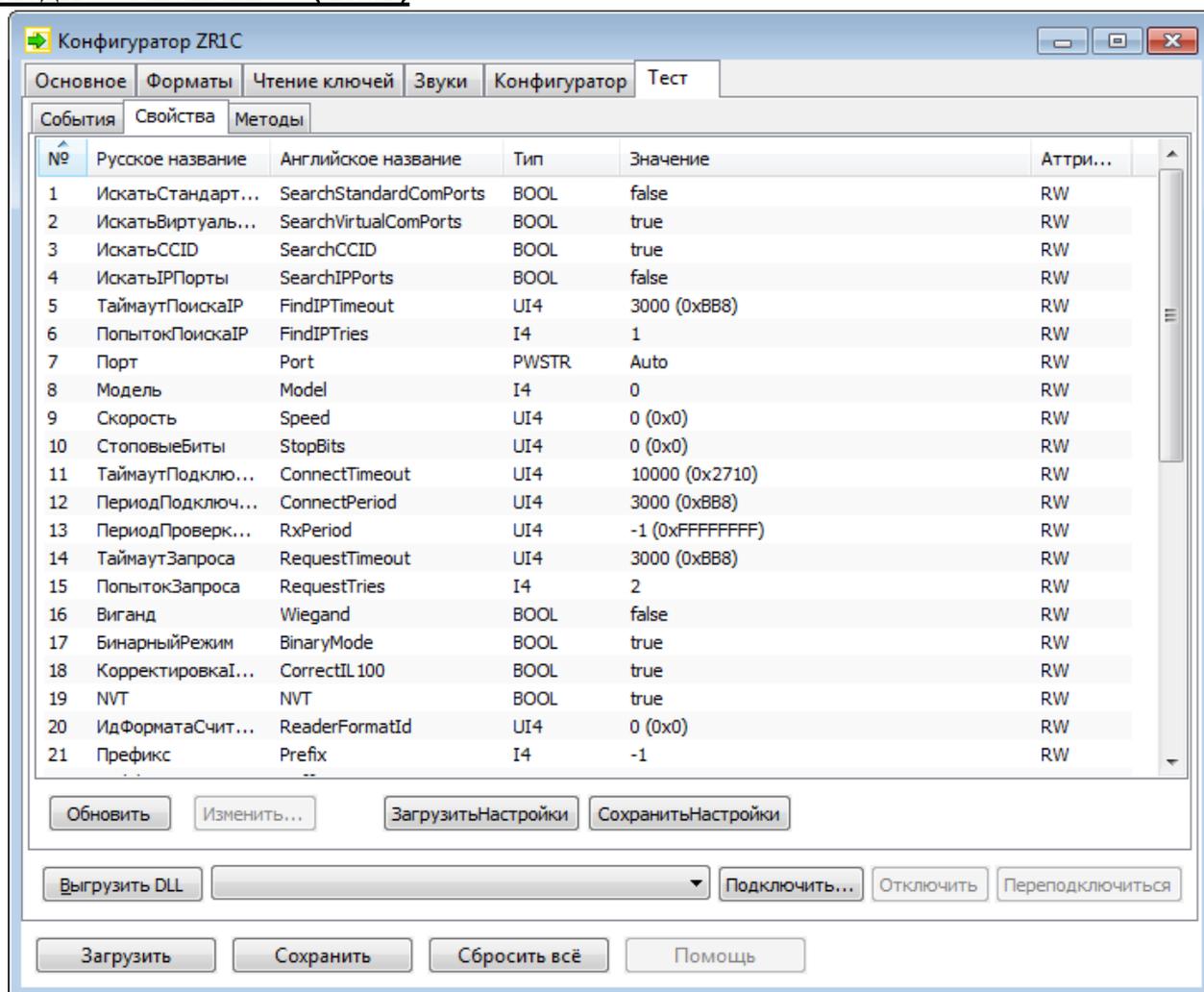
Основные типы событий:

- [ДанныеКарты \(TracksData\)](#) – передан номера ключа;
- [ОшибкаДрайвера \(DriverError\)](#) – произошла ошибка при работе со считывателем.

Дополнительные типы событий (включаются на вкладке «[Основное](#)»):

- [УстройствоПодключено \(DeviceConnect\)](#) – считыватель подключён;
- [УстройствоОтключено \(DeviceDisconnect\)](#) – считыватель отключён;
- [НетКарты \(NoCard\)](#) – карта удалена из поля считывателя;
- [Комманда \(Command\)](#) – завершён метод, запущенный в фоновом режиме.

Кнопка «**Очистить**» – очищает содержимое таблицы.

Вкладка "Свойства" (Тест)

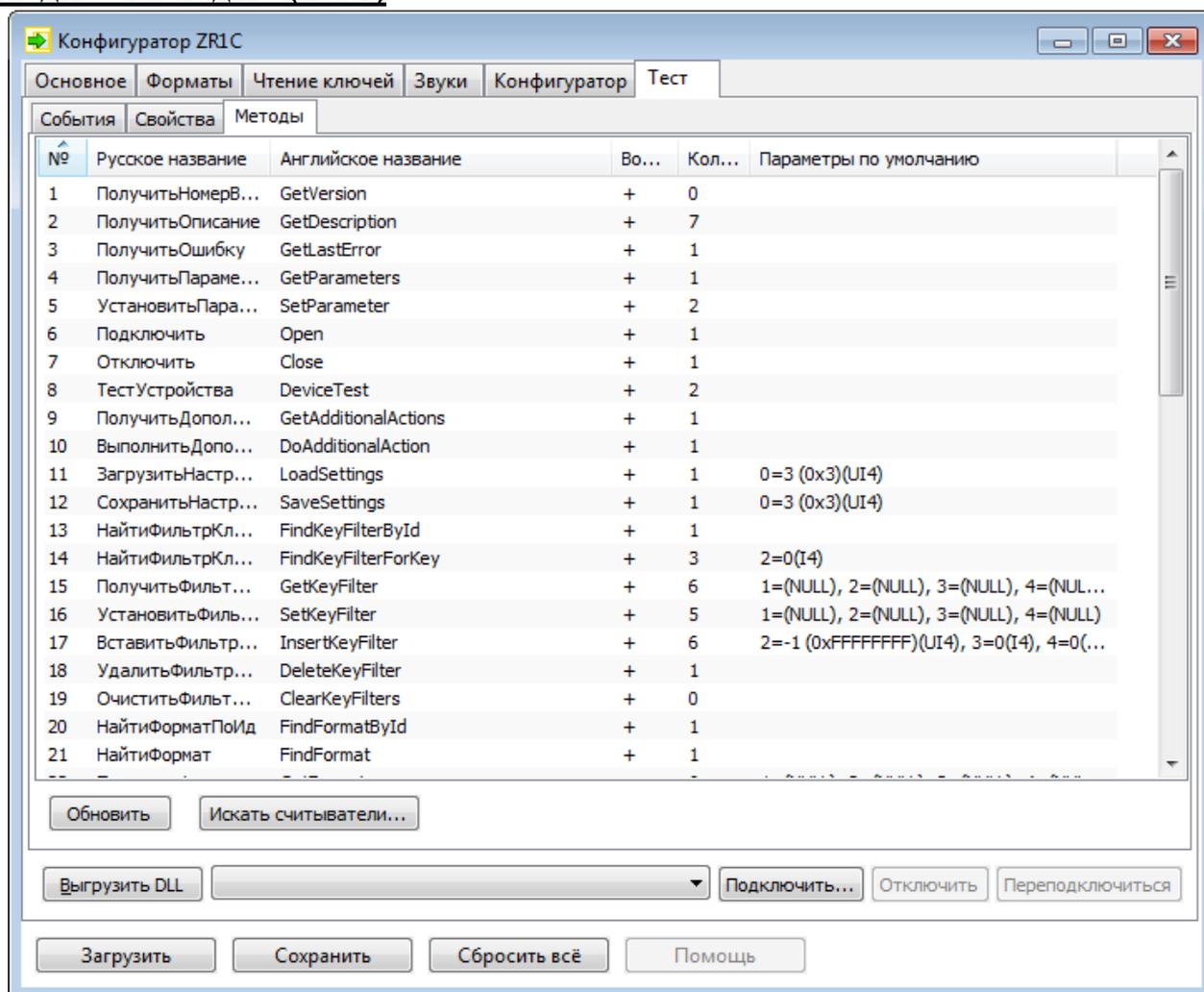
Эта страница предназначена для тестирования свойств компоненты без использования IC. В таблице отображаются список свойств компоненты.

Кнопка **"Обновить"** – обновляет список.

Кнопка **"Изменить..."** – открывает окно "Свойство" для изменения значения свойства, выделенного в таблице.

Кнопка **"ЗагрузитьНастройки"** – вызывает метод [ЗагрузитьНастройки \(LoadSettings\)](#) компоненты.

Кнопка **"СохранитьНастройки"** – вызывает метод [СохранитьНастройки \(SaveSettings\)](#) компоненты.

Вкладка "Методы" (Тест)

Эта страница предназначена для тестирования методов компоненты без использования IC. В таблице отображаются список методов компоненты.

Дв.клик ЛКМ по методу в таблице открывает окно "Метод", позволяющее вызывать метод компоненты.

Кнопка **"Обновить"** – обновляет список.

Кнопка **"Искать считыватели"** – открывает окно "Поиск считывателей" с результатами поиска, осуществляемого с помощью методов [НайтиСледующий \(FindNext\)](#) и [ЗавершитьПоиск \(FindClose\)](#).

Компонента ZR1C

Компонента ZR1C предназначена для передачи номеров ключей от считывателя IronLogic в ПО "IC: Предприятие".

Свойства компоненты

Свойство	Описание
Параметры поиска считывателей	
ИскатьСтандартныеComПорты (SearchStandardComPorts)	True, искать стандартные COM порты (RS-232).

Искать Виртуальные Com Порты (SearchVirtualComPorts)	True, искать виртуальные COM порты (USB).
Искать CCID (SearchCCID)	True, искать считыватели CCID.
Искать IP Порты (SearchIPPorts)	True, искать IP-конвертеры.
Таймаут Поиска IP (FindIPTimeout)	Тайм-аут ожидания ответа на запрос поиска IP-конвертеров по UDP в миллисекундах.
Попыток Поиска IP (FindIPTries)	Количество попыток запроса поиска IP-конвертеров.

Параметры подключения к считывателю

Модель (Model)	Модель считывателя.
Порт (Port)	Имя порта считывателя (com-порт или адрес порта ip-конвертера).
Скорость (Speed)	Максимальная скорость считывателя.
Стоповые Биты (StopBits)	Количество стоповых битов для com-порта.
Таймаут Подключения (ConnectTimeout)	Тайм-аут подключения по TCP в миллисекундах.
Период Подключения (ConnectPeriod)	Период подключения в миллисекундах – интервал времени между попытками открыть порт.
Период Проверки Порта (RxPeriod)	Период проверки входящих данных порта в миллисекундах.
Таймаут Запроса (RequestTimeout)	Тайм-аут запроса в миллисекундах.
Попыток Запроса (RequestTries)	Количество попыток запроса.
Виганд (Wiegand)	True, если считыватель подключен к Z-2 Base по Wiegand, иначе – по Dallas.
Бинарный Режим (BinaryMode)	True, использовать бинарный режим, если возможно. Используется для Z-2 USB MF , Z-2 MF-I , Matrix III Net , Matrix V .
Разные Коды IL100 (DifferentIL100)	True, передаёт разные коды кнопок брелка IL-100 независимо от прошивки считывателя (Z-2 EHR , Matrix V), иначе – одинаковые.
NVT (NVT)	True, по TCP данные передаются по протоколу NVT.
ИД Формата Считывателя (ReaderFormatID)	Id формата, используемый считывателем.
Префикс (Prefix)	Префикс для разделения дорожек в ВнешнееСобытие.
Суффикс (Suffix)	Суффикс для разделения дорожек в ВнешнееСобытие.

Настройки лога

Уровень Лога (LogLevel)	Уровень лога.
Макс Размер Лога (MaxLogSize)	Максимальный размер лог файла (МБайт).

Настройки чтения ключей

Типы Ключей (KeyTypes)	Типы ключей, разрешённые для передачи в 1С.
Таймаут Удерживания Ключа (KeyHoldTimeout)	Тайм-аут удерживания ключа (в мс)
Удерживать MifareC (HoldMifareC)	True, при обнаружении ключа типа Mifare Classic сканирование приостанавливается.

УдерживатьMifareP (HoldMifareP)	True, при обнаружении ключа типа Mifare Plus с уровнем безопасности SL3 сканирование приостанавливается.
УдерживатьMifareUL (HoldMifareUL)	True, при обнаружении ключа типа Mifare Ultralight сканирование приостанавливается.
УдерживатьTemic (HoldTemic)	True, при обнаружении ключа типа Temic сканирование приостанавливается.
ПараметрыСканированияTemic (TemicScanParameters)	Параметры сканирования карт Temic .
АвтоMifareC (AutoMifareC)	True, читать номер из блока Mifare Classic .
АвтоMifareP (AutoMifareP)	True, читать номер из блока Mifare Plus SL3 .
АвтоMifareUL (AutoMifareUL)	True, читать номер из страницы Mifare Ultralight .
АвтоTemic (AutoTemic)	True, читать номер из блока Temic .

Настройки звуковых уведомлений

ЗвукВключен (SoundEnabled)	True, если звуковые уведомления включены.
ЗвукПередачи (SoundTransfer)	Путь к звуковому файлу, воспроизводимому после успешной передачи номера ключа.
ЗвукОшибкиПередачи (SoundTransferError)	Путь к звуковому файлу, воспроизводимому при ошибке передачи номера ключа.
ЗвукПодключения (SoundConnections)	Путь к звуковому файлу, воспроизводимому при подключении к считывателю.
ЗвукОтключения (SoundDisconnections)	Путь к звуковому файлу, воспроизводимому при отключении от считывателя.
ЗвукНайденКлюч (SoundFoundKey)	Путь к звуковому файлу, воспроизводимому при подносе карты к считывателю.
ЗвукПотерянКлюч (SoundLostKey)	Путь к звуковому файлу, воспроизводимому при удалении карты от считывателя.

Текущие параметры чтения/записи Mifare

КлючАвторизацииMC (AuthorizationKeyMC)	Текущий явный ключ авторизации Mifare Classic в 16-ричном формате.
КлючАвторизацииMP (AuthorizationKeyMP)	Текущий явный ключ авторизации Mifare Plus SL3 в 16-ричном формате.

Другие

НастройкиИзменены (SettingsModified)	True, настройки изменены.
КоличествоФорматов (FormatCount)	Количество форматов в списке.
КоличествоФильтровКлючей (KeyFilterCount)	Количество фильтров ключей в списке.
ИДФорматаПоУмолчанию (DefaultFormatID)	Id формата по умолчанию, используемого, когда Id формата считывателя = 0 и не найден формат для ключа, соответствующего одному из фильтров ключей.

Свойство ИскатьСтандартныеComПорты (SearchStandardComPorts)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), искать стандартные COM порты (RS-232), к которым подключены считыватели. По умолчанию = Ложь (False).

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство ИскатьВиртуальныеComПорты (SearchVirtualComPorts)

Использование: Чтение и запись.

Тип: Булево (Boolean). Внутренний тип: BOOL.

Описание:

Истина (True), искать виртуальные COM порты (USB), к которым подключены считыватели. По умолчанию = Истина (True).

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство ИскатьCCID (SearchCCID)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), искать считыватели CCID. По умолчанию = Истина (True).

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство ИскатьIPПорты (SearchIPPorts)

Использование: Чтение и запись.

Тип: Булево (Boolean). Внутренний тип: BOOL.

Описание:

Истина (True), искать IP-конвертеры, к которым подключены считыватели. По умолчанию = Ложь (False).

Поиск осуществляется отправкой широковещательного запроса в порт UDP 9000.

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство ТаймаутПоискаIP (FindIPTimeout)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Тайм-аут ожидания ответа на запрос поиска IP-конвертеров по UDP в миллисекундах. По умолчанию = 3000.

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство ПопытокПоискаIP (FindIPTries)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Количество попыток запроса поиска IP-конвертеров. По умолчанию = 1.

Используется методом [НайтиСледующий \(FindNext\)](#) и при открытии порта с именем "Авто".

Свойство Модель (Model)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Модель считывателя. Если = 0, то определяется автоматически. По умолчанию = 0.

Значение	Описание
0	Авто
1	Z-2 USB, Z-2 Rd All
2	Matrix III Rd-All

3	Z-2 USB MF
4	Matrix III Net
5	CP-Z2-MF
6	Z-2 EHR
7	Z-2 Base
8	Matrix V
9	RF-1996
10	Z-2 Rd All
11	Z-2 USB MFI
12	Z-2 MF CCID

Используется методом [Подключить \(Open\)](#).

Свойство Порт (Port)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Имя порта считывателя (COM-порт или адрес порта IP-конвертера). Если = "Авто" ("Auto") (название не чувствительно к регистру символов), то определяется автоматически. По умолчанию = "Auto".

Если используется сетевая прошивка Matrix III Rd-All, то имя порта должно быть в формате "COM3\123", где "COM3" – порт конвертера, "123" – серийный номер (с/н) считывателя.

Используется методом [Подключить \(Open\)](#).

Свойство Скорость (Speed)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Максимальная скорость считывателя. Если = 0, то определяется автоматически. По умолчанию = 0.

Позволяет ограничить скорость работы со считывателем.

Используется методом [Подключить \(Open\)](#).

Свойство СтоповыеБиты (StopBits)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Количество стоповых битов для COM-порта. По умолчанию = 0.

Значение	Описание
0	1 стоповый бит.
1	1,5 стоповых бита.
2	2 стоповых бита.

Используется методом [Подключить \(Open\)](#).

Свойство ТаймаутПодключения (ConnectTimeout)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Тайм-аут подключения по TCP в миллисекундах. По умолчанию =10000.

Используется методом [Подключить \(Open\)](#).

Свойство ПериодПодключения (ConnectPeriod)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Период подключения в миллисекундах, т.е. интервал времени между попытками открыть порт.

Если = -1, то связь автоматически не восстанавливается. По умолчанию =3000.

Используется методом [Подключить \(Open\)](#).

Свойство ПериодПроверкиПорта (RxPeriod)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Период проверки входящих данных порта в миллисекундах. Если = -1, то входящие данные проверяются только по сигналу от порта. По умолчанию = -1.

Используется методом [Подключить \(Open\)](#).

Свойство ТаймаутЗапроса (RequestTimeout)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Тайм-аут запроса в миллисекундах. По умолчанию = 3000.

Используется методом [Подключить \(Open\)](#).

Свойство ПопытокЗапроса (RequestTries)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Количество попыток запроса. По умолчанию = 2.

Используется методом [Подключить \(Open\)](#).

Свойство Виганд (Wiegand)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), если считыватель подключен по Wiegand, иначе – по Dallas. По умолчанию = Ложь (False).

Используется для [Z-2 Base](#).

Используется методом [Подключить \(Open\)](#).

Свойство БинарныйРежим (BinaryMode)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, если считыватель подключен по Wiegand, иначе – по Dallas. По умолчанию = Истина (True).

Используется для [Z-2 USB MF](#), [Z-2 MF-I](#), [Matrix III Net](#), [CP-Z2-MF](#).

Используется методом [Подключить \(Open\)](#).

Свойство РазныеКодыIL100 (DifferentIL100)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, передаёт разные коды кнопок брелка IL-100 независимо от прошивки считывателя (Z-2 EHR, Matrix V), иначе – одинаковые.

Формат	Кнопка 1		Кнопка 2	
	Em+код	16-ричный	Em+код	16-ричный
Одинаковые	Cod433 Fix[FD] 083,64879	0000FD53FD6F	Cod433 Fix[FB] 083,64879	0000FB53FD6F
Разные	Cod433 Fix[F5] 211,64879	0000F5D3FD6F	Cod433 Fix[F5] 179,64879	0000F5B3FD6F

По умолчанию = Истина (True).

Используется для [Z-2 EHR](#), [Matrix V](#).

Используется методом [Подключить \(Open\)](#).

Свойство NVT (NVT)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), по TCP данные передаются по протоколу NVT (англ. Network Virtual Terminal, [RFC2217](#)), иначе – как простой текст. По умолчанию = Истина (True).

Используется для считывателей, подключенных к IP конвертеру Z-397.

Используется методом [Подключить \(Open\)](#).

Свойство ИДФорматаСчитывателя (ReaderFormatID)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

ID формата, используемый считывателем. По умолчанию = 0.

Если = 0, то при поднесении ключа формат выбирается автоматически с помощью фильтров ключей.

Используется методом [Подключить \(Open\)](#).

Свойство Префикс (Prefix)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Префикс для разделения дорожек в ВнешнееСобытие (ExternalEvent). По умолчанию = -1. Свойство требуется поддержки технологии "1С совместимо".

Возможные значения:

Значение	Описание	Значение	Описание	Значение	Описание	Значение	Описание
-1	<NONE>						
0	0 (NUL)	32	32 (SPACE)	64	64 (@)	96	96 ()
1	1 (SOH)	33	33 (!)	65	65 (A)	97	97 (a)
2	2 (STX)	34	34 (")	66	66 (B)	98	98 (b)
3	3 (ETX)	35	35 (#)	67	67 (C)	99	99 (c)
4	4 (EOT)	36	36 (\$)	68	68 (D)	100	100 (d)
5	5 (ENQ)	37	37 (%)	69	69 (E)	101	101 (e)
6	6 (ACK)	38	38 (&)	70	70 (F)	102	102 (f)
7	7 (BEL)	39	39 (')	71	71 (G)	103	103 (g)
8	8 (BS)	40	40 (())	72	72 (H)	104	104 (h)
9	9 (TAB)	41	41 ())	73	73 (I)	105	105 (i)
10	10 (LF)	42	42 (*)	74	74 (J)	106	106 (j)
11	11 (VT)	43	43 (+)	75	75 (K)	107	107 (k)
12	12 (FF)	44	44 (,)	76	76 (L)	108	108 (l)
13	13 (CR)	45	45 (-)	77	77 (M)	109	109 (m)

14	14 (SO)	46	46 (.)	78	78 (N)	110	110 (n)
15	15 (SI)	47	47 (/)	79	79 (O)	111	111 (o)
16	16 (DLE)	48	48 (0)	80	80 (P)	112	112 (p)
17	17 (DC1)	49	49 (1)	81	81 (Q)	113	113 (q)
18	18 (DC2)	50	50 (2)	82	82 (R)	114	114 (r)
19	19 (DC3)	51	51 (3)	83	83 (S)	115	115 (s)
20	20 (DC4)	52	52 (4)	84	84 (T)	116	116 (t)
21	21 (NAK)	53	53 (5)	85	85 (U)	117	117 (u)
22	22 (SYN)	54	54 (6)	86	86 (V)	118	118 (v)
23	23 (ETB)	55	55 (7)	87	87 (W)	119	119 (w)
24	24 (CAN)	56	56 (8)	88	88 (X)	120	120 (x)
25	25 (EM)	57	57 (9)	89	89 (Y)	121	121 (y)
26	26 (SUB)	58	58 (:)	90	90 (Z)	122	122 (z)
27	27 (ESC)	59	59 (;)	91	91 (I)	123	123 (I)
28	28 (FS)	60	60 (<)	92	92 (\)	124	124 (I)
29	29 (GS)	61	61 (=)	93	93 (J)	125	125 (J)
30	30 (RS)	62	62 (>)	94	94 (^)	126	126 (~)
31	31 (US)	63	63 (?)	95	95 (_)	127	127 (DEL)

Используется методом [Подключить \(Open\)](#).

Свойство Суффикс (Suffix)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Суффикс для разделения дорожек в ВнешнееСобытие (ExternalEvent). По умолчанию = 13. Свойство требуется поддержки технологии "IC совместимо".

Возможные значения:

Значение	Описание	Значение	Описание	Значение	Описание	Значение	Описание
-1	<NONE>						
0	0 (NUL)	32	32 (SPACE)	64	64 (@)	96	96 (`)
1	1 (SOH)	33	33 (!)	65	65 (A)	97	97 (a)
2	2 (STX)	34	34 (")	66	66 (B)	98	98 (b)
3	3 (ETX)	35	35 (#)	67	67 (C)	99	99 (c)
4	4 (EOT)	36	36 (\$)	68	68 (D)	100	100 (d)
5	5 (ENQ)	37	37 (%)	69	69 (E)	101	101 (e)
6	6 (ACK)	38	38 (&)	70	70 (F)	102	102 (f)
7	7 (BEL)	39	39 (')	71	71 (G)	103	103 (g)
8	8 (BS)	40	40 (())	72	72 (H)	104	104 (h)
9	9 (TAB)	41	41 ())	73	73 (I)	105	105 (i)
10	10 (LF)	42	42 (*)	74	74 (J)	106	106 (j)
11	11 (VT)	43	43 (+)	75	75 (K)	107	107 (k)
12	12 (FF)	44	44 (,)	76	76 (L)	108	108 (l)
13	13 (CR)	45	45 (-)	77	77 (M)	109	109 (m)
14	14 (SO)	46	46 (.)	78	78 (N)	110	110 (n)
15	15 (SI)	47	47 (/)	79	79 (O)	111	111 (o)

16	16 (DLE)	48	48 (0)	80	80 (P)	112	112 (p)
17	17 (DC1)	49	49 (1)	81	81 (Q)	113	113 (q)
18	18 (DC2)	50	50 (2)	82	82 (R)	114	114 (r)
19	19 (DC3)	51	51 (3)	83	83 (S)	115	115 (s)
20	20 (DC4)	52	52 (4)	84	84 (T)	116	116 (t)
21	21 (NAK)	53	53 (5)	85	85 (U)	117	117 (u)
22	22 (SYN)	54	54 (6)	86	86 (V)	118	118 (v)
23	23 (ETB)	55	55 (7)	87	87 (W)	119	119 (w)
24	24 (CAN)	56	56 (8)	88	88 (X)	120	120 (x)
25	25 (EM)	57	57 (9)	89	89 (Y)	121	121 (y)
26	26 (SUB)	58	58 (:)	90	90 (Z)	122	122 (z)
27	27 (ESC)	59	59 (;)	91	91 (I)	123	123 (})
28	28 (FS)	60	60 (<)	92	92 (\)	124	124 ()
29	29 (GS)	61	61 (=)	93	93 (J)	125	125 (})
30	30 (RS)	62	62 (>)	94	94 (^)	126	126 (~)
31	31 (US)	63	63 (?)	95	95 (_)	127	127 (DEL)

Используется методом [Подключить \(Open\)](#).

Свойство Уровень Лога (LogLevel)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Уровень лога. По умолчанию = 0.

Путь к лог-файлу "%APPDATA%\IronLogic\ZR1C\zr1c.log".

Значение	Описание
0	Лог выключен.
1	Неожиданные ошибки, которых быть не должно.
2	Ошибки. Показывает ошибки системных функций, исключения.
3	Предупреждения. Показывает возможные проблемы, которые не являются ошибками.
4	Уведомления. Показывает полезную информацию, в основном успехи.
5	Отладочные сообщения. Показывает шаги программы, получаемые и отправляемые данные.
6	Подробные отладочные сообщения. Показывает каждую мелочь.

Используется методом [Подключить \(Open\)](#).

Свойство МаксРазмерЛога (MaxLogSize)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Максимальный размер лог файла (МБайт). Если = -1, то не ограничено. По умолчанию = -1.

Путь к лог-файлу "%APPDATA%\IronLogic\ZR1C\zr1c.log". Если размер файла превысил максимум, то лог файл копируется (с заменой) в "ZR1C.log.old", и затем лог файл очищается.

Используется методом [Подключить \(Open\)](#).

Свойство ТипыКлючей (KeyTypes)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Типы ключей, разрешённые для передачи в 1С. По умолчанию = 4194303 (0x3FFFFFF).

Бит	Тип ключа
0	Неизвестный
1	Em-Marine
2	Hid
3	iCode
4	Cod433 Fix
5	Cod433
6	радиобрелок CAME
7	Dallas
8	Temic (T5557)
9	Mifare UltraLight
10	Mifare Classic 1K
11	Mifare Classic 2K
12	Mifare Classic 4K
13	Mifare Plus
14	Mifare Plus 1K
15	Mifare Plus 2K
16	Mifare Plus 4K
17	Smart MX with Mifare 1K
18	Smart MX with Mifare 4K
19	Mifare DESFire
20	Mifare ProX
21	Mifare Mini

Свойство ТаймаутУдерживанияКлюча (KeyHoldTimeout)

Использование: Чтение и запись.

Тип: Число (Number).

Описание:

Тайм-аут удерживания ключа в миллисекундах. Если = -1, то сканирование приостанавливается пока не будет вызван метод [ПродолжитьСканирование \(ContinueScanning\)](#). По умолчанию = 4194303 (0x3FFFFFF).

Позволяет приостановить на время цикл сканирования ключей, чтобы ускорить работу методов чтения/записи памяти ключей:

- [ЧитатьMifareC \(ReadMifareC\)](#)
- [ПисатьMifareC \(WriteMifareC\)](#)
- [ЧитатьMifareP \(ReadMifareP\)](#)
- [ПисатьMifareP \(WriteMifareP\)](#)
- [ЧитатьMifareUL \(ReadMifareUL\)](#)
- [ПисатьMifareUL \(WriteMifareUL\)](#)
- [ЧитатьTemic \(ReadTemic\)](#)
- [ПисатьTemic \(WriteTemic\)](#)

Типы удерживаемых ключей после обнаружения нового ключа выбираются с помощью свойств:

- [УдерживатьMifareC \(HoldMifareC\)](#)
- [УдерживатьMifareP \(HoldMifareP\)](#)
- [УдерживатьMifareUL \(HoldMifareUL\)](#)
- [УдерживатьTemic \(HoldTemic\)](#)

Свойство УдерживатьMifareC (HoldMifareC)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), при обнаружении ключа типа [Mifare Classic](#) сканирование приостанавливается. По умолчанию = Ложь (False).

Время, на которое приостанавливается сканирование, устанавливается в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Свойство УдерживатьMifareP (HoldMifareP)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), при обнаружении ключа типа [Mifare Plus](#) с уровнем безопасности SL3 сканирование приостанавливается. По умолчанию = Ложь (False).

Время, на которое приостанавливается сканирование, устанавливается в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Свойство УдерживатьMifareUL (HoldMifareUL)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), при обнаружении ключа типа [Mifare Ultralight](#) сканирование приостанавливается. По умолчанию = Ложь (False).

Время, на которое приостанавливается сканирование, устанавливается в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Свойство УдерживатьTemic (HoldTemic)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), при обнаружении ключа типа [Temic](#) сканирование приостанавливается. По умолчанию = Ложь (False).

Время, на которое приостанавливается сканирование, устанавливается в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Свойство ПараметрыСканированияTemic (TemicScanParameters)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Строка с параметрами сканирования карт Temic, содержащая список запросов, разделенных символом ';', каждый запрос содержит список параметров (тип, скорость, задержка), разделенных символом ',':

1. Тип – =1 с терминатором, =2 без терминатора. Если =255, то вызывается команда считывателя "reset", которая ненадолго снимает генерацию 125кГц, чтобы считыватель перешел в нормальный режим работы;
2. Скорость – =0 стандарт div64, =1 div32;
3. Задержка – интервал времени в миллисекундах между запросами.

По умолчанию = "2,0,0;1,1,0;2,1,0;1,0,0;255,0,1000;".

Используется методами [ИскатьTemic \(FindTemic\)](#), [ЧитатьTemic \(ReadTemic\)](#), [ПисатьTemic \(WriteTemic\)](#).

Свойство АвтоMifareC (AutoMifareC)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, автоматически читать номер из блока Mifare Classic при поднесении ключа. По умолчанию = Ложь (False).

Свойство АвтоMifareP (AutoMifareP)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, автоматически читать номер из блока Mifare Plus SL3 при поднесении ключа. По умолчанию = Ложь (False).

Свойство АвтоMifareUL (AutoMifareUL)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, автоматически читать номер из страницы Mifare Ultralight при поднесении ключа. По умолчанию = Ложь (False).

Свойство АвтоTemic (AutoTemic)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

True, автоматически читать номер из блока Temic при поднесении ключа. По умолчанию = Ложь (False).

Свойство ЗвукВключен (SoundEnabled)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), если звуковые уведомления включены. По умолчанию = Ложь (False).

Свойство ЗвукПередачи (SoundTransfer)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому после успешной передачи номера ключа. По умолчанию = "".

Свойство ЗвукОшибкиПередачи (SoundTransferError)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому при ошибке передачи номера ключа. По умолчанию = "".

Свойство ЗвукПодключения (SoundConnections)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому при подключении к считывателю. По умолчанию = "".

Свойство ЗвукОтключения (SoundDisconnections)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому при отключении от считывателя. По умолчанию = "".

Свойство ЗвукНайденКлюч (SoundFoundKey)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому при подносе карты к считывателю. По умолчанию = "".

Свойство ЗвукПотерянКлюч (SoundLostKey)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Путь к звуковому файлу, воспроизводимому при удалении карты от считывателя. По умолчанию = "".

Свойство КлючАвторизацииMC (AuthorizationKeyMC)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Текущий явный ключ авторизации Mifare Classic в 16-ричном формате. По умолчанию = "FFFFFFFFFFFF".

Например, "A1B2C3D4E5F6", где младший байт слева, или "0xF6E5D4C3B2A1", где младший байт справа.

Используется методами [ReadMifareC](#) и [WriteMifareC](#).

Свойство НастройкиИзменены (SettingsModified)

Использование: Чтение и запись.

Тип: Булево (Boolean).

Описание:

Истина (True), настройки изменены.

Для сохранения настроек используйте метод [СохранитьНастройки \(SaveSettings\)](#).

Свойство КлючАвторизацииMP (AuthorizationKeyMP)

Использование: Чтение и запись.

Тип: Строка (String).

Описание:

Текущий явный ключ авторизации Mifare Plus SL3 в 16-ричном формате. По умолчанию = "FFFFFFFFFFFFFFFFFFFFFFFFFFFF".

Например, "A1B2C3D4E5F6", где младший байт слева, или "0xF6E5D4C3B2A1", где младший байт справа.

Используется методами [ReadMifareP](#) и [WriteMifareP](#).

Свойство КоличествоФорматов (FormatCount)

Использование: Чтение.

Тип: Число (Number).

Описание:

Количество форматов в списке.

Свойство КоличествоФильтровКлючей (KeyFilterCount)**Использование:** Чтение.**Тип:** Число (Number).**Описание:**

Количество фильтров ключей в списке. Максимум 31.

Свойство ИДФорматаПоУмолчанию (DefaultFormatID)**Использование:** Чтение и запись.**Тип:** Число (Number).**Описание:**

Id формата по умолчанию, используемого когда Id формата считывателя = 0 и не найден формат для ключа, соответствующего одному из фильтров ключей. По умолчанию = 0.

[схема выбора формата для ключа](#)**Методы компоненты**

Метод	Описание
Методы для поддержки технологии "1С совместимо"	
ПолучитьНомерВерсии (GetVersion)	Возвращает номер версии этой компоненты.
ПолучитьОписание (GetDescription)	Возвращает информацию о компоненте, такую как название и описание, поддерживаемый тип оборудования.
ПолучитьОшибку (GetLastError)	Возвращает код и описание последней произошедшей ошибки.
ПолучитьПараметры (GetParameters)	Возвращает список параметров настройки компоненты и их типы, значения по умолчанию и возможные значения.
УстановитьПараметр (SetParameter)	Устанавливает значение параметра по имени.
Подключить (Open)	Подключает оборудование с текущими значениями параметров, установленных методом УстановитьПараметр (SetParameter) . Возвращает идентификатор подключенного экземпляра устройства.
Отключить (Close)	Отключает оборудование.
ТестУстройства (DeviceTest)	Выполняет пробное подключение и опрос устройства с текущими значениями параметров, установленными методом УстановитьПараметр (SetParameter) . При успешном выполнении подключения в описании возвращается информация об устройстве.
ПолучитьДополнительныеДействия (GetAdditionalActions)	Получает список действий, которые будут отображаться как дополнительные пункты меню в форме настройки оборудования, доступной администратору. Если действий не предусмотрено, возвращает пустую строку.
ВыполнитьДополнительноеДействие (DoAdditionalAction)	Команда на выполнение дополнительного действия с определенными именем.

Управление настройками компоненты

ЗагрузитьНастройки (LoadSettings)	Загружает настройки компоненты из реестра.
СохранитьНастройки (SaveSettings)	Сохраняет настройки компоненты в реестр.
НайтиФильтрКлючейПоИД (FindKeyFilterByID)	Ищет позицию в списке фильтров ключей по Id.

НайтиФильтрКлючейДляКлюча (FindKeyFilterForKey)	Ищет позицию в списке фильтров ключей для указанного ключа (по типу ключа и длине номера ключа).
ПолучитьФильтрКлючей (GetKeyFilter)	Возвращает параметры фильтра ключей по позиции в списке.
УстановитьФильтрКлючей (SetKeyFilter)	Устанавливает параметры фильтра ключей по позиции в списке.
ВставитьФильтрКлючей (InsertKeyFilter)	Вставляет новый фильтр ключей в указанную позицию списка.
УдалитьФильтрКлючей (DeleteKeyFilter)	Удаляет фильтр ключей в указанной позиции списка.
ОчиститьФильтрыКлючей (ClearKeyFilters)	Удаляет все фильтры ключей в списке.
НайтиФорматПоИД (FindFormatByID)	Ищет позицию в списке форматов по Id.
НайтиФормат (FindFormat)	Ищет позицию в списке форматов по названию.
ПолучитьФормат (GetFormat)	Возвращает параметры формата по позиции в списке.
УстановитьФормат (SetFormat)	Устанавливает параметры формата по позиции в списке.
ВставитьФормат (InsertFormat)	Создает новый формат и вставляет в указанную позицию списка.
УдалитьФормат (DeleteFormat)	Удаляет формат в указанной позиции списка.
ОчиститьФорматы (ClearFormats)	Удаляет все форматы в списке.
ПроверитьФормат (CheckFormat)	Проверяет шаблон и параметры формата.
Форматировать (Format)	Форматирует номер ключа и возвращает строку.
Поиск считывателей	
НайтиСледующий (FindNext)	Ищет следующий считыватель и возвращает информацию о нём.
ЗавершитьПоиск (FindClose)	Завершает поиск считывателей, начатый функцией НайтиСледующий (FindNext) .
Работа со считывателем	
ОбновитьПараметрыСчитывателя (UpdateReaderParameters)	Копирует параметры из указанного считывателя в текущие параметры подключения считывателя, и наоборот.
ПолучитьСостояние (GetStatus)	Возвращает состояние подключения к считывателю.
Переподключиться (Reconnect)	Запускает переподключение считывателя (не ждет завершения операции).
ПолучитьИнфоСчитывателя (GetReaderInfo)	Возвращает информацию о считывателе, зарегистрированном методом Подключить (Open) .
ПолучитьИнфоКлюча (GetKeyInfo)	Возвращает номер карты в поле считывателя.
ПродолжитьСканирование (ContinueScanning)	Возобновляет сканирование карт в поле считывателя.
ПрерватьКоманду (BreakCommand)	Прерывает выполнение команды считывателя.
ПолучитьСостояниеКоманды (GetCommandStatus)	Возвращает состояние выполнения последней команды считывателя.
ПолучитьРезультатКоманды (GetCommandResult)	Возвращает результат последней команды считывателя.

Для считывателей [Z-2 RD ALL](#), [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Rd-All](#), [Matrix III Net](#), [CP-Z2-MF](#)

ЧитатьMifareUL (ReadMifareUL)	Читает одну или несколько страниц из Mifare Ultralight .
ПисатьMifareUL (WriteMifareUL)	Записывает одну или несколько страниц в Mifare Ultralight .

Для считывателей [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#)

ЧитатьMifareC (ReadMifareC)	Читает один или несколько блоков из Mifare Classic и из Mifare Plus SL1 .
ПисатьMifareC (WriteMifareC)	Записывает один или несколько блоков в Mifare Classic и в Mifare Plus SL1 .

Для считывателей [Z-2 MF-I](#)

ЧитатьMifareP (ReadMifareP)	Читает один или несколько блоков из Mifare Plus SL3 .
ПисатьMifareP (WriteMifareP)	Записывает один или несколько блоков в Mifare Plus SL3 .

Для считывателей [Z-2 USB MF](#), [Z-2 MF-I](#), [Matrix III Net](#), [CP-Z2-MF](#)

ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader)	Записывает один или несколько ключей авторизации Mifare в память считывателя.
Сигнализировать (Signal)	Сигнализирует с помощью индикаторов и звука считывателя.

Для считывателей [Z-2 RD ALL](#), [Z-2 EHR](#)

ИнициализироватьTemic (InitializeTemic)	Записывает стандартную конфигурацию в Temic (0x40801400).
ИскатьTemic (FindTemic)	Ищет карту Temic .
ЧитатьTemic (ReadTemic)	Читает данные из блоков карты Temic .
ПисатьTemic (WriteTemic)	Пишет данные в блоки карты Temic .

Сервисные

ДобавитьВЛог (AddLog)	Добавляет сообщение в лог файл "ZR1C.log".
ОчиститьЛог (ClearLog)	Удаляет лог файл "ZR1C.log".

Метод ПолучитьНомерВерсии (GetVersion)

Синтаксис:

`ПолучитьНомерВерсии()`

Возвращаемое значение:

Тип: Строка.

Строка с номером версии компоненты ZR1C – "<версия zr1c.dll>".

Описание:

Возвращает номер версии этой компоненты.

Метод ПолучитьОписание (GetDescription)**Синтаксис:**

```

ПолучитьОписание(
[out] <НазваниеКомпоненты>,
[out] <ОписаниеКомпоненты>,
[out] <ТипОборудования>,
[out] <РевизияИнтерфейса>,
[out] <ИнтеграционнаяБиблиотека>,
[out] <ОсновнойДрайверУстановлен>,
[out] <ПолучитьURLСкачивания>)

```

Параметры:**<НазваниеКомпоненты>**

Тип: Строка. Название компоненты – "ZR1C";

<ОписаниеКомпоненты>

Тип: Строка. Описание компоненты – "Драйвер для подключения устройств ввода";

<ТипОборудования>

Тип: Строка. Строка, определяющая тип оборудования – "СчитывательМагнитныхКарт";

<РевизияИнтерфейса>

Тип: Число. Поддерживаемая версия требований для данного типа оборудования – 1005;

<ИнтеграционнаяБиблиотека>

Тип: Булево. Флаг возвращает, является ли компонент интеграционной библиотекой драйвера или самостоятельным драйвером – Ложь (False);

<ОсновнойДрайверУстановлен>

Тип: Булево. Для интеграционной библиотеки возвращает флаг установки основной поставки драйвера – Истина (True);

<ПолучитьURLСкачивания>

Тип: Строка. Адрес страницы сайта производителя, по которому доступна ссылка для скачивания основной поставки компоненты – "http://www.ironlogic.ru".

Возвращаемое значение:

Тип: Число.

Текущее состояние подключения:

Значение	Описание
0	Отключен
1	Подключен
2	В процессе подключения

Описание:

Возвращает информацию о компоненте, такую как название и описание, поддерживаемый тип оборудования.

Метод ПолучитьОшибку (GetLastError)**Синтаксис:**

```

ПолучитьОшибку(
[out] <ОписаниеОшибки>)

```

Параметры:

<ОписаниеОшибки>

Тип: Строка. Описание ошибки.

Возвращаемое значение:

Тип: Число. [Код ошибки](#).

Описание:

Возвращает код и описание последней произошедшей ошибки.

Метод ПолучитьПараметры (GetParameters)

Синтаксис:

```
ПолучитьПараметры(
[out] <ТаблицаПараметров>)
```

Параметры:

<ТаблицаПараметров>

Тип: Строка. Таблица параметров в формате xml.

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Возвращает список параметров настройки компоненты и их типы, значения по умолчанию и возможные значения.

Метод УстановитьПараметр (SetParameter)

Синтаксис:

```
УстановитьПараметр(
[in] <Имя>,
[in] <Значение>)
```

Параметры:

<Имя>

Тип: Строка. Имя параметра:

Название	Тип	Описание
Порт (Port)	Строка	Имя порта считывателя
Модель (Model)	Число	Модель считывателя.
Скорость (Speed)	Число	Предельная скорость считывателя.
СтоповыеБиты (StopBits)	Число	Количество стоповых битов com-порта.
ТаймаутПодключения (ConnectTimeout)	Число	Тайм-аут подключения по TCP в миллисекундах.
ПериодПодключения (ConnectPeriod)	Число	Период попыток подключения в миллисекундах.
ПериодПроверкиПорта (RxPeriod)	Число	Период проверки входящих порта в миллисекундах.
ТаймаутЗапроса (ReqTimeout)	Число	Тайм-аут ожидания ответа на запрос в миллисекундах.
ПопытокЗапроса (ReqTries)	Число	Количество попыток послать запрос.
Виганд (Wiegand)	Булево	True, если считыватель подключен по Wiegand, иначе – по Dallas. Для Z-2 Base.
БинарныйРежим (BinaryMode)	Булево	True, использовать бинарный режим, если возможно.

ИДФорматаСчитывателя (ReaderFormatID)	Число	Id формата, назначаемое считывателю.
Префикс (Prefix)	Число	Префикс для разделения дорожек в ВнешнееСобытие.
Суффикс (Suffix)	Число	Суффикс для разделения дорожек в ВнешнееСобытие.

<Значение>

Тип: Вариант, зависит от <Имя>. Значение параметра.

Возвращаемое значение:

Тип: Булево. Истина (True), значение установлено, иначе – нет.

Описание:

Устанавливает значение параметра по имени.

Метод Подключить (Open)**Синтаксис:**

```
Подключить(
[out] <ИДУстройства>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Булево. Истина (True), считыватель подключен, иначе – нет.

Описание:

Подключает оборудование с текущими значениями параметров, установленных методом [УстановитьПараметр \(SetParameter\)](#). Возвращает идентификатор подключенного экземпляра устройства.

Примечание:

Имя порта задается с помощью метода `SetParameter("Port", "COM3")` или с помощью свойства [Порт \(Port\)](#).

Если используется сетевая прошивка Matrix III Rd-All, то имя порта должно быть в формате "COM3\123", где "COM3" – порт конвертера, "123" – серийный номер (с/н) считывателя.

Метод Отключить (Close)**Синтаксис:**

```
Отключить(
[in] <ИДУстройства>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Отключает оборудование.

Метод ТестУстройства (DeviceTest)

Синтаксис:

```
ТестУстройства(  
[out] <Описание>,  
[out] <АктивированДемоРежим>)
```

Параметры:

<Описание>

Тип: Строка. Описание результата выполнения теста – информационные строки считывателя;

<АктивированДемоРежим>

Тип: Строка. Описание ограничений демонстрационного режима – пустая строка.

Возвращаемое значение:

Тип: Булево. Истина (True), тестирование прошло успешно, иначе – не успешно.

Описание:

Выполняет пробное подключение и опрос устройства с текущими значениями параметров, установленными методом [УстановитьПараметр \(SetParameter\)](#). При успешном выполнении подключения в описании возвращается информация об устройстве.

Метод ПолучитьДополнительныеДействия (GetAdditionalActions)

Синтаксис:

```
ПолучитьДополнительныеДействия(  
[out] <ТаблицаДействий>)
```

Параметры:

<ТаблицаДействий>

Тип: Строка. Список дополнительных действий – пустая строка.

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Получает список действий, которые будут отображаться как дополнительные пункты меню в форме настройки оборудования, доступной администратору. Если действий не предусмотрено, возвращает пустую строку.

Метод ВыполнитьДополнительноеДействие (DoAdditionalAction)

Синтаксис:

```
ВыполнитьДополнительноеДействие(  
[in] <ИмяДействия>)
```

Параметры:

<ИмяДействия>

Тип: Строка. Имя действия.

Возвращаемое значение:

Тип: Булево. Всегда Ложь (False).

Описание:

Команда на выполнение дополнительного действия с определенными именем.

Метод ЗагрузитьНастройки (LoadSettings)

Синтаксис:

ЗагрузитьНастройки(
[in] <Флаги> = 3)

Параметры:

<Флаги>

Тип: Число. Маска параметров. По умолчанию 3.

Бит	Описание
0	Параметры подключения к считывателю и все остальные параметры, кроме списка форматов и списка фильтров ключей.
1	Список форматов и список фильтров ключей.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Описание:

Загружает настройки компоненты из реестра.

Метод СохранитьНастройки (SaveSettings)

Синтаксис:

СохранитьНастройки(
[in] <Флаги> = 0)

Параметры:

<Флаги>

Тип: Число. Маска параметров. Если = 0, то сохраняются только изменённые. По умолчанию 0.

Бит	Описание
0	Основные настройки. Все параметры, кроме списка форматов и списка фильтров ключей.
1	Список форматов и список фильтров ключей.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Описание:

Сохраняет настройки компоненты в реестр.

Примечание

С помощью свойства [НастройкиИзменены \(SettingsModified\)](#) можно определить изменены ли настройки.

Метод НайтиФильтрКлючейПоИД (FindKeyFilterByID)

Синтаксис:

НайтиФильтрКлючейПоИД

```
(  
  [in] <ИД>  
)
```

Параметры:

<ИД>

Тип: Число. Идентификатор фильтра ключей.

Возвращаемое значение:

Тип: Число. Позиция в списке. Если не найден, то возвращает -1.

Описание:

Ищет позицию в списке фильтров ключей по ID.

Метод НайтиФильтрКлючейДляКлюча (FindKeyFilterForKey)

Синтаксис:

НайтиФильтрКлючейДляКлюча(

```
[in] <ТипКлюча>,  
[in] <ДлинаНомера>,  
[in] <НачальнаяПозиция> = 0  
)
```

Параметры:

<ТипКлюча>

Тип: Число. Тип ключа;

<ДлинаНомера>

Тип: Число. Длина номера ключа;

<НачальнаяПозиция>

Тип: Число. Начальная позиция, с которой начинается поиск.

Возвращаемое значение:

Тип: Число. Позиция в списке. Если не найден, то возвращает -1.

Описание:

Ищет позицию в списке фильтров ключей для указанного ключа (по типу ключа и длине номера ключа).

Метод ПолучитьФильтрКлючей (GetKeyFilter)

Синтаксис:

ПолучитьФильтрКлючей(

```
[in] <Позиция>,  
[out] <Название>,  
[out] <ТипыКлючей>,  
[out] <ДлинаНомера>,  
[out] <ИДФормата>,  
[out] <ИД>)
```

Параметры:**<Позиция>**

Тип: Число. Позиция в списке фильтров ключей;

<Название>

Тип: Строка. Название фильтра;

<ТипыКлючей>

Тип: Число. Маска типов ключей;

<ДлинаНомера>

Тип: Число. Длина номера ключа;

<ИДФормата>

Тип: Число. ID формата, используемого для форматирования номера ключа;

<ИД>

Тип: Число. ID фильтра ключей.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат успешно получен, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.

Описание:

Возвращает параметры фильтра ключей по позиции в списке.

Метод УстановитьФильтрКлючей (SetKeyFilter)**Синтаксис:****УстановитьФильтрКлючей(**

```
[in] <Позиция>,
[in] <Название> = Null,
[in] <ТипыКлючей> = Null,
[in] <ДлинаНомера> = Null,
[in] <ИДФормата> = Null)
```

Параметры:**<Позиция>**

Тип: Число. Позиция в списке фильтров ключей;

<Название>

Тип: Строка. Название фильтра. Если = Null, то не изменять;

<ТипыКлючей>

Тип: Число. Маска типов ключей. Если = Null, то не изменять;

<ДлинаНомера>

Тип: Число. Длина номера ключа. Если = Null, то не изменять;

<ИДФормата>

Тип: Число. ID формата, используемого для форматирования номера ключа. Если = Null, то не изменять.

Возвращаемое значение:

Тип: Булево. Истина (True), выполнено успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.
ZR1C_E_INVALIDKEYFILTERNAME	-2147220928 (0x80040240)	Некорректное название фильтра.
ZR1C_E_KEYFILTERNAMEEXIST	-2147220927 (0x80040241)	Фильтр с указанным именем уже существует.

Описание:

Устанавливает параметры фильтра ключей по позиции в списке.

Метод ВставитьФильтрКлючей (InsertKeyFilter)**Синтаксис:****ВставитьФильтрКлючей(**

```
[in,out] <Позиция>,
[in] <Название>,
[in] <ТипыКлючей> = -1,
[in] <ДлинаНомера> = 0,
[in] <ИДФормата> = 0,
[out] <ИД> = Null)
```

Параметры:**<Позиция>**

Тип: Число. Позиция вставки. Если позиция вне диапазона, то добавляет в конец списка;

<Название>

Тип: Строка. Уникальное название фильтра.

<ТипыКлючей>

Тип: Число. Маска типов ключей. Если = -1, то все типы;

<ДлинаНомера>

Тип: Число. Длина номера ключа. Если = 0, то длина игнорируется.

<ИДФормата>

Тип: Число. ID формата, используемого для форматирования номера ключа. Если = 0, то нет формата. Если = Null, то не изменять;

<ИД>

Тип: Число. ID фильтра ключей.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат успешно изменен, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_INVALIDKEYFILTERNAME	-2147220928 (0x80040240)	Некорректное название фильтра.
ZR1C_E_KEYFILTERNAMEEXIST	-2147220927 (0x80040241)	Фильтр с указанным именем уже существует.
ZR1C_E_KEYFILTERLIMIT	-2147220926 (0x80040242)	Достигнут предел количества фильтров ключей, максимум 31 (0x1F).

Описание:

Вставляет новый фильтр ключей в указанную позицию списка.

Метод УдалитьФильтрКлючей (DeleteKeyFilter)**Синтаксис:**

УдалитьФильтрКлючей(
[in] <Позиция>)

Параметры:

<Позиция>

Тип: Число. Позиция в списке фильтров ключей.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.

Описание:

Удаляет фильтр ключей в указанной позиции списка.

Метод ОчиститьФильтрыКлючей (ClearKeyFilters)

Синтаксис:

ОчиститьФильтрыКлючей()

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Удаляет все фильтры ключей в списке.

Метод НайтиФорматПоИД (FindFormatByID)

Синтаксис:

НайтиФорматПоИД(
[in] <ID>
)

Параметры:

<ID>

Тип: Число. Идентификатор формата.

Возвращаемое значение:

Тип: Число. Позиция в списке. Если не найден, то возвращает -1.

Описание:

Ищет позицию в списке форматов по ID.

Метод НайтиФормат (FindFormat)

Синтаксис:

НайтиФормат(
[in] <Название>)

Параметры:

<Название>

[in] Название формата.

Возвращаемое значение:

Тип: Число. Позиция в списке. Если не найден, то возвращает -1.

Описание:

Ищет позицию в списке форматов по названию.

Метод ПолучитьФормат (GetFormat)**Синтаксис:**

```
ПолучитьФормат(
  [in] <Позиция>,
  [out] <Название>,
  [out] <Шаблон>,
  [out] <Параметры>,
  [out] <Комментарий>,
  [out] <ID>)
```

Параметры:**<Позиция>**

Тип: Число. Позиция в списке форматов;

<Название>

Тип: Строка. Название формата;

<Шаблон>

Тип: Строка. Шаблон;

<Параметры>

Тип: Строка. Параметры;

<Комментарий>

Тип: Строка. Комментарий;

<ID>

Тип: Число. ID формата.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат успешно получен, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x80000000B)	Индекс вне диапазона.

Описание:

Возвращает параметры формата по позиции в списке.

Метод УстановитьФормат (SetFormat)**Синтаксис:**

```
УстановитьФормат(
  [in] <Позиция>,
  [in] <Название> = Null,
  [in] <Шаблон> = Null,
  [in] <Параметры> = Null,
```

[in] <Комментарий> = Null)

Параметры:**<Позиция>**

Тип: Число. Позиция в списке форматов;

<Название>

Тип: Строка. Уникальное название формата. Если = Null, то не изменять;

<Шаблон>

Тип: Строка. Шаблон. Если = Null, то не изменять;

<Параметры>

Тип: Строка. Параметры. Если = Null, то не изменять;

<Комментарий>

Тип: Строка. Текст комментария. Если = Null, то не изменять.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат успешно изменен, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.
ZR1C_E_INVALIDFORMATNAME	-2147220940 (0x80040234)	Некорректное название формата.
ZR1C_E_FORMATNAMEEXIST	-2147220939 (0x80040235)	Формат с указанным именем уже существует.

Описание:

Изменяет формат по позиции в списке.

Метод ВставитьФормат (InsertFormat)**Синтаксис:**

```

ВставитьФормат(
  [in,out] <Позиция>,
  [in] <Название>,
  [in] <Шаблон>,
  [in] <Параметры>,
  [in] <Комментарий> = Null,
  [out] <ID> = Null)

```

Параметры:**<Позиция>**

Тип: Число. Позиция вставки. Если позиция вне диапазона, то добавляет в конец списка;

<Название>

Тип: Строка. Уникальное название формата. Значение не может быть пустым;

<Шаблон>

Тип: Строка. Шаблон. Значение не может быть пустым;

<Параметры>

Тип: Строка. Параметры;

<Комментарий>

Тип: Строка. Комментарий. Если = Null, то не изменять.

<ID>

Тип: Число. ID формата.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат успешно создан, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_INVALIDFORMATNAME	-2147220940 (0x80040234)	Некорректное название формата.
ZR1C_E_FORMATNAMEEXIST	-2147220939 (0x80040235)	Формат с указанным именем уже существует.
ZR1C_E_FORMATLIMIT	-2147220925 (0x80040243)	Достигнут предел количества форматов, максимум 4294967295 (0xffffffff).

Описание:

Создает новый формат и вставляет в указанную позицию списка.

Метод УдалитьФормат (DeleteFormat)

Синтаксис:

```
УдалитьФормат(
[in] <Позиция>)
```

Параметры:

<Позиция>

Тип: Число. Позиция в списке.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.

Описание:

Удаляет формат в указанной позиции списка.

Метод ОчиститьФорматы (ClearFormats)

Синтаксис:

```
ОчиститьФорматы()
```

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Удаляет все форматы в списке.

Метод ПроверитьФормат (CheckFormat)

Синтаксис:

```
ПроверитьФормат(
[in] <Шаблон>,
[in] <Параметры>,
```

[out] <ПозицияОшибки> = Null)

Параметры:

<Шаблон>

Тип: Строка. Шаблон;

<Параметры>

Тип: Строка. Параметры;

<ПозицияОшибки>

Тип: Число. Позиция ошибки.

Возвращаемое значение:

Тип: Булево. Истина (True), если формат корректный, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_NOTENOUGHPARAMS	-2147220938 (0x80040236)	Недостаточно параметров.
ZR1C_E_SPECNOTSUPPORTED	-2147220937 (0x80040237)	Спецификатор шаблона не поддерживается.
ZR1C_E_UNKNOWNKEYWORD	-2147220936 (0x80040238)	Неизвестное ключевое слово.
ZR1C_E_ENDEXPRESSION	-2147220935 (0x80040239)	Неожиданный конец выражения.
ZR1C_E_UNKNOWNSYMBOL	-2147220934 (0x8004023A)	Неизвестный символ.
ZR1C_E_LBRACKET	-2147220933 (0x8004023B)	Ожидается '[' или '['!.
ZR1C_E_RBRACKET	-2147220932 (0x8004023C)	Ожидается ')' или ']'!.
ZR1C_E_NUMBERSYNTAX	-2147220931 (0x8004023D)	Ошибка синтаксиса числа.

Описание:

Проверяет шаблон и параметры формата.

Метод Форматировать (Format)

Синтаксис:

```
Форматировать(
  [out] <Результат>,
  [in] <Шаблон>,
  [in] <Параметры>,
  [in] <ТипКлюча>,
  [in] <НомерКлюча>)
```

Параметры:

<Результат>

Тип: Строка. Строка результата форматирования;

<Шаблон>

Тип: Строка. [Шаблон](#) формата;

<Параметры>

Тип: Строка. [Параметры](#) формата;

<ТипКлюча>

Тип: Число. Тип ключа;

<НомерКлюча>

Тип: Число, Строка, ДвоичныеДанные. Байты номер ключа. Если тип = "Строка", то строка в 16-ричном формате.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Описание:

Форматирует номер ключа и возвращает строку.

Метод НайтиСледующий (FindNext)**Синтаксис:**

```
НайтиСледующий(
[out] <ИДУстройства>,
[out] <Модель>,
[out] <С/н>,
[out] <ВерсияПрошивки>,
[out] <ИнфоСчитывателя>,
[in] <Тайм-аут> = -1)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Модель>

Тип: Число. Модель;

<С/н>

Тип: Число. Серийный номер считывателя;

<ВерсияПрошивки>

Тип: Число. Версия прошивки;

<ИнфоСчитывателя>

Тип: Строка. Информационные строки считывателя;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не найдётся считыватель или пока поиск не завершится.

Возвращаемое значение:

Тип: Булево. Истина (True), считыватель найден, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_FINDTIMEOUT	-2147220944 (0x80040230)	Тайм-аут поиска считывателей. Возникает, если значение тайм-аута ожидания отличается от -1.
ZR1C_E_NOMOREREADERS	-2147220943 (0x80040231)	Нет больше считывателей.

Описание:

Ищет следующий считыватель и возвращает информацию о нём.

Метод ЗавершитьПоиск (FindClose)**Синтаксис:**

```
ЗавершитьПоиск()
```

Возвращаемое значение:

Тип: Булево. Всегда Истина (True).

Описание:

Завершает поиск считывателей, начатый функцией [FindNext / НайтиСледующий](#).

Метод ОбновитьПараметрыСчитывателя (UpdateReaderParameters)

Синтаксис:

```
ОбновитьПараметрыСчитывателя(
  [in] <ИДУстройства>,
  [in] <Сохранить> = True
)
```

Параметры:

<ИДУстройства>

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сохранить>

Тип: Булево. Истина (True), применяет [текущие параметры](#) подключения считывателя к считывателю <ИДУстройства>, иначе копирует параметры считывателя <ИДУстройства> в текущие параметры подключения считывателя;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Обновляет параметры указанного считывателя или [текущие параметры](#) подключения считывателю.

Метод ПолучитьСостояние (GetStatus)

Синтаксис:

```
ПолучитьСостояние(
  [in] <ИДУстройства>)
```

Параметры:

<ИДУстройства>

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Число. Текущее состояние подключения:

Код состояния	Описание
0	Не зарегистрирован методом Подключить (Open)
1	Отключен
2	В процессе подключения
3	Подключен

Описание:

Возвращает состояние подключения к считывателю.

Метод Переподключиться (Reconnect)**Синтаксис:**

```
Переподключиться(
[in] <ИДУстройства>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Запускает переподключение считывателя (не ждет завершения операции).

Метод ПолучитьИнфоСчитывателя (GetReaderInfo)**Синтаксис:**

```
ПолучитьИнфоСчитывателя(
[in] <ИДУстройства>,
[out] <Модель>,
[out] <С/н>,
[out] <ВерсияПрошивки>,
[out] <ИнфоСчитывателя>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Модель>

Тип: Число. Модель считывателя;

<С/н>

Тип: Число. Серийный номер;

<ВерсияПрошивки>

Тип: Число. Версия прошивки;

<ИнфоСчитывателя>

Тип: Строка. Информационные строки считывателя.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Возвращает информацию о считывателе, зарегистрированном методом [Подключить \(Open\)](#). Информация может быть недостоверной, если компонента не успела подключиться к считывателю после вызова [Подключить \(Open\)](#), или если при подключении возникла ошибка, в этих случаях метод [ПолучитьСостояние \(GetStatus\)](#) будет возвращать значение, не равное "3" ("подключен").

Метод ПолучитьИнфоКлюча (GetKeyInfo)**Синтаксис:**

```
ПолучитьИнфоКлюча(
[in] <ИДУстройства>,
[in] <Позиция> = 0,
[out] <ФорматированныйНомер> = Null,
[out] <ТипКлюча> = Null,
[out] <НомерКлюча> = Null,
[out] <НомерКлюча2> = Null,
[out] <Параметр1> = Null,
[out] <Параметр2> = Null)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Позиция>

Тип: Число. Индекс ключа в списке ключей, найденных в поле считывателя;

<ФорматированныйНомер>

Тип: Строка. Форматированный номер ключа.

<ТипКлюча>

Тип: Число. Тип ключа:

Значение	Описание
0	Не было ключа.
1	Em-Marine
2	Hid
3	iCode
4	Cod433 Fix
5	Cod433
6	радиобрелок CAME
7	Dallas
8	Temic (T5557)
9	Mifare UltraLight
10	Mifare Classic 1K
11	Mifare Classic 2K
12	Mifare Classic 4K
13	Mifare Plus
14	Mifare Plus 1K

15	Mifare Plus 2K
16	Mifare Plus 4K
17	Smart MX with Mifare 1K
18	Smart MX with Mifare 4K
19	Mifare DESFire
20	Mifare ProX
21	Mifare Mini

<НомерКлюча>

Тип: ДвоичныеДанные. Байты UID ключа.

<НомерКлюча2>

Тип: ДвоичныеДанные. Байты, автоматически считанные из памяти карты.

<Параметр1>

Тип: Число. Зависит от типа ключа.

Тип ключа	Описание
Temic	Параметр сканирования Temic
Mifare Plus	Версия: =0 Mifare Plus S, =1 Mifare Plus X.

<Параметр2>

Тип: Число. Зависит от типа ключа.

Тип ключа	Описание
Mifare Plus	Уровень SL (Security level): 0-3. Если = -1, то не известно.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован методом Подключить (Open) .
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет ключа.

Описание:

Возвращает номер карты в поле считывателя.

Метод ПродолжитьСканирование (ContinueScanning)**Синтаксис:**

```
ПродолжитьСканирование(
[in] <ИДУстройства>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Возобновляет сканирование карт в поле считывателя.

Метод ПрерватьКоманду (BreakCommand)**Синтаксис:**

ПрерватьКоманду(
[in] <ИДУстройства>)

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя).

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Прерывает выполнение текущей команды считывателя, запущенной одним из методов перечисленных [здесь](#).

Метод ПолучитьСостояниеКоманды (GetCommandStatus)**Синтаксис:**

ПолучитьСостояниеКоманды(
[in] <ИДУстройства>,
[out] <ТипКоманды> = Null,
[out] <КодРезультата> = Null,
[out] <ТекущийШаг> = Null,
[out] <ВсегоШагов> = Null,
[in] <Тайм-аут> = -1)

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<ТипКоманды>

Тип: Число. Тип команды:

Значение	Описание
0	Нет команды.
1	Метод ЧитатьMifareUL (ReadMifareUL) .
2	Метод ПисатьMifareUL (WriteMifareUL) .
3	Метод ЧитатьMifareC (ReadMifareC) .

4	Метод ПисатьMifareC (WriteMifareC) .
5	Метод ЧитатьMifareP (ReadMifareP) .
6	Метод ПисатьMifareP (WriteMifareP) .
7	Метод ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader) .
8	Метод Сигнализировать (Signal) .
9	Метод ИнициализироватьTemic (InitializeTemic) .
10	Метод ИскатьTemic (FindTemic) .
11	Метод ЧитатьTemic (ReadTemic) .
12	Метод ПисатьTemic (WriteTemic) .

<КодРезультата>

Тип: Число. Код результата команды. Если = E_PENDING -2147483638 (0x8000000A), то команда все еще выполняется;

<ТекущийШаг>

Тип: Число. Текущая позиция прогресса (для команд чтения/записи Mifare = количеству считанных/записанных страниц/блоков);

<ВсегоШагов>

Тип: Число. Максимальная позиция прогресса.

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока команда не будет выполнена или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Возвращает состояние выполнения последней команды считывателя.

Метод ПолучитьРезультатКоманды (GetCommandResult)**Синтаксис:****ПолучитьРезультатКоманды(**

```
[in] <ИДУстройства>,
[in] <ТипКоманды> = Null,
[out] <Данные> = Null,
[out] <Параметр> = Null)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<ТипКоманды>

Тип: Число. Тип команды:

Значение	Описание
0	Нет команды.
1	Метод ЧитатьMifareUL (ReadMifareUL) .
2	Метод ПисатьMifareUL (WriteMifareUL) .
3	Метод ЧитатьMifareC (ReadMifareC) .
4	Метод ПисатьMifareC (WriteMifareC) .
5	Метод ЧитатьMifareP (ReadMifareP) .
6	Метод ПисатьMifareP (WriteMifareP) .
7	Метод ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader) .
8	Метод Сигнализировать (Signal) .
9	Метод ИнициализироватьTemic (InitializeTemic) .
10	Метод ИскатьTemic (FindTemic) .
11	Метод ЧитатьTemic (ReadTemic) .
12	Метод ПисатьTemic (WriteTemic) .

<Данные>

Тип: ДвоичныеДанные. Считанные данные карты.

<Параметр>

Тип: Число. Значение параметра зависит от типа команды:

Тип команды	Описание параметра
3, 4	Позиция найденного ключа авторизации в памяти считывателя при авторизации по маске при чтении/записи Mifare Classic.
5, 6	Позиция найденного ключа авторизации в памяти считывателя при авторизации по маске при чтении/записи Mifare Plus SL3.
10, 11	Параметр сканирования Temic.
остальные	Не используется.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Описание:

Возвращает результат последней команды считывателя.

Метод ЧитатьMifareUL (ReadMifareUL)**Синтаксис:**

```

ЧитатьMifareUL(
  [in] <ИДУстройства>,
  [in] <Адрес>,
  [out] <Данные>,
  [in] <КоличествоСтраниц> = 1,
  [in] <Тайм-аут> = -1)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Адрес>

Тип: Число. Номер первой страницы (от 0 до 15, см. [структуру памяти Mifare Ultralight](#));

<Данные>

Тип: ДвоичныеДанные. Считанные данные страниц (каждая страница по 4 байта);

<КоличествоСтраниц>

Тип: Число. Количество страниц;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут считаны страницы или пока не произойдет ошибка.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем считаны страницы, то функция вернёт E_PENDING и чтение страниц продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить чтение используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилось ли чтение используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#). Чтобы получить считанные данные используйте функцию [ПолучитьРезультатКоманды \(GetCommandResult\)](#).

Описание:

Читает одну или несколько страниц из [Mifare Ultralight](#).

Поддерживают считыватели:

- в бинарном режиме: [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#);
- в ascii-режиме: [Z-2 RD_ALL](#), [Z-2 USB MF](#), [Z-2 MF CCID](#), [Matrix III Rd-All](#).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [Таймаут Удерживания Ключа \(KeyHoldTimeout\)](#).

Метод Писать MifareUL (WriteMifareUL)**Синтаксис:**

```
ПисатьMifareUL(
[in] <ИДУстройства>,
[in] <Адрес>,
[in] <Данные>,
[in] <КоличествоСтраниц> = 1,
[in] <Тайм-аут> = -1)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Адрес>

Тип: Число. Номер первой страницы (от 0 до 15, см. [структуру памяти Mifare Ultralight](#));

<Данные>

Тип: ДвоичныеДанные, Число, Строка. Данные страниц (каждая страница по 4 байта). Если тип "Строка", то строка с байтами в 16-ричном формате (например, "A1B2C3D4", младший байт слева, или "0xD4C3B2A1", младший байт справа), пробелы игнорируются. Если размер данных < (4 * Количество страниц), то недостающие данные заполняются нулями;

<КоличествоСтраниц>

Тип: Число. Количество страниц;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут записаны страницы или пока не произойдет ошибка.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.

Константа	Значение	Описание
ZR1C_E_KEYWRITELOCK	-2147220963 (0x8004021D)	Страница заблокирована от перезаписи.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем записаны страницы, то функция вернёт E_PENDING и запись страниц продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить запись используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилось ли запись используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#).

Описание:

Записывает одну или несколько страниц в [Mifare Ultralight](#).

Поддерживают считыватели:

- в бинарном режиме: [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#);
- в ascii-режиме: [Z-2 RD_ALL](#), [Z-2 USB MF](#), [Z-2 MF CCID](#), [Matrix III Rd-All](#).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ЧитатьMifareC (ReadMifareC)

Синтаксис:

```

ЧитатьMifareC(
  [in] <ИДУстройства>,
  [in] <Сектор>,
  [in] <Блок>,
  [out] <Данные>,
  [in] <КоличествоБлоков> = 1,
  [in] <КлючиСчитывателя> = 0,
  [in] <КлючБ> = False,
  [in] <ПропуститьПрицепы> = False,
  [in] <Тайм-аут> = -1,
  [out] <ПозицияКлюча> = Null)

```

Параметры:

<ИДУстройства>

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сектор>

Тип: Число. Номер первого сектора (см. структуру памяти [Mifare 1K](#) и [Mifare 4K](#));

<Блок>

Тип: Число. Номер первого блока;

<Данные>

Тип: ДвоичныеДанные. Считанные данные (каждый блок по 16 байт);

<КоличествоБлоков>

Тип: Число. Количество блоков;

Тип карты	Секторов	Всего блоков	Всего блоков без прицепов
Mifare Mini	5	20	15
Mifare 1K	16	64	48
Mifare 2K	32	128	96
Mifare 4K	40	256	216

<КлючиСчитывателя>

Тип: Число. Маска авторизации по ключам в памяти считывателя. Если = 0, то используется явный ключ авторизации, который устанавливается в свойстве [КлючАвторизацииМС \(AuthorizationKeyМС\)](#);

<КлючБ>

Тип: Булево. Авторизоваться по ключу Б (иначе – по ключу А);

<ПропуститьПрицепы>

Тип: Булево. Пропустить блоки-прицепы;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут считаны блоки или пока не произойдет ошибка;

<ПозицияКлюча>

Тип: Число. Позиция найденного ключа авторизации при авторизации по маске ключей считывателя. Если = -1, то не найден.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_AUTHORIZE	-2147220965 (0x8004021B)	Ошибка авторизации ключа.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем считаны блоки, то функция вернёт E_PENDING и чтение блоков продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить чтение используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилось ли чтение используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#). Чтобы получить считанные данные используйте функцию [ПолучитьРезультатКоманды \(GetCommandResult\)](#).

Описание:

Читает один или несколько блоков из [Mifare Classic](#) и из Mifare Plus SL1.
Поддерживают считыватели [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#) в бинарном режиме (см. свойство [БинарныйРежим](#)).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ПисатьMifareC (WriteMifareC)**Синтаксис:**

```

ПисатьMifareC(
  [in] <ИДУстройства>,
  [in] <Сектор>,
  [in] <Блок>,
  [in] <Данные>,
  [in] <КоличествоБлоков> = 1,
  [in] <КлючиСчитывателя> = 0,
  [in] <КлючБ> = False,
  [in] <ПропуститьПрицепы> = False,
  [in] <Тайм-аут> = -1,
  [out] <ПозицияКлюча> = Null)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сектор>

Тип: Число. Номер первого сектора (1К: от 0..15; 4К: от 0 до 39, см. структуру памяти [Mifare 1К](#) и [Mifare 4К](#));

<Блок>

Тип: Число. Номер первого блока;

<Данные>

Тип: ДвоичныеДанные. Данные для записи (каждый блок по 16 байт);

<КоличествоБлоков>

Тип: Число. Количество блоков:

Тип карты	Секторов	Всего блоков	Всего блоков без прицепов
Mifare Mini	5	20	15
Mifare 1К	16	64	48
Mifare 2К	32	128	96
Mifare 4К	40	256	216

<КлючиСчитывателя>

Тип: Число. Маска авторизации по ключам в памяти считывателя. Если = 0, то используется явный ключ авторизации, который устанавливается в свойстве [КлючАвторизацииМС \(AuthorizationKeyMC\)](#);

<КлючБ>

Тип: Булево. Авторизоваться по ключу Б (иначе – по ключу А);

<ПропуститьПрицепы>

Тип: Булево. Пропустить блоки-прицепы. В блоке-прицепе хранятся настройки доступа к сектору: ключ авторизации А и Б, биты доступа. Если в блок-прицеп записать некорректное значение – весь сектор становится недоступным навсегда;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут записаны блоки или пока не произойдет ошибка;

<ПозицияКлюча>

Тип: Число. Позиция найденного ключа авторизации при авторизации по маске ключей считывателя. Если = -1, то не найден.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_AUTHORIZE	-2147220965 (0x8004021B)	Ошибка авторизации ключа.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем записаны блоки, то функция вернёт E_PENDING и запись блоков продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить запись используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилась ли запись используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#).

Описание:

Записывает один или несколько блоков в [Mifare Classic](#) и в Mifare Plus SL1. Поддерживают считыватели [Z-2 USB MF](#), [Z-2 MF-I](#), [Z-2 MF CCID](#), [Matrix III Net](#), [CP-Z2-MF](#) в бинарном режиме (см. свойство [БинарныйРежим](#)).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ЧитатьMifareP (ReadMifareP)

Синтаксис:

```

ЧитатьMifareP(
  [in] <ИДУстройства>,
  [in] <Сектор>,
  [in] <Блок>,
  [out] <Данные>,
  [in] <КоличествоБлоков> = 1,
  [in] <КлючиСчитывателя> = 0,
  [in] <КлючБ> = False,
  [in] <ПропуститьПрицепы> = False,
  [in] <ОткрытыйТекст> = True,
  [in] <Тайм-аут> = -1,
  [out] <ПозицияКлюча> = Null)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сектор>Тип: Число. Номер первого сектора (1К: от 0..15; 4К: от 0 до 39, см. структуру памяти [Mifare 1К](#) и [Mifare 4К](#));**<Блок>**

Тип: Число. Номер первого блока;

<Данные>

Тип: ДвоичныеДанные. Считанные данные (каждый блок по 16 байт);

<КоличествоБлоков>

Тип: Число. Количество блоков:

Тип карты	Секторов	Всего блоков	Всего блоков без прицепов
Mifare 1К	16	64	48
Mifare 2К	32	128	96
Mifare 4К	40	256	216

<КлючиСчитывателя>Тип: Число. Маска авторизации по ключам в памяти считывателя. Если = 0, то используется явный ключ авторизации, который устанавливается в свойстве [КлючАвторизацииMP \(AuthorizationKeyMP\)](#);**<КлючБ>**

Тип: Булево. Авторизоваться по ключу Б (иначе – по ключу А);

<ПропуститьПрицепы>

Тип: Булево. Пропустить блоки-прицепы;

<ОткрытыйТекст>

Тип: Булево. Истина (True), открытый текст, иначе – закрытый текст;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут считаны блоки или пока не произойдет ошибка;

<ПозицияКлюча>

Тип: Число. Позиция найденного ключа авторизации при авторизации по маске ключей считывателя. Если = -1, то не найден.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_AUTHORIZE	-2147220965 (0x8004021B)	Ошибка авторизации ключа.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем считаны блоки, то функция вернёт E_PENDING и чтение блоков продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить чтение используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилось ли чтение используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#). Чтобы получить считанные данные используйте функцию [ПолучитьРезультатКоманды \(GetCommandResult\)](#).

Описание:

Читает один или несколько блоков из [Mifare Plus SL3](#).

Поддерживает считыватель [Z-2 MF-I](#) в бинарном режиме (см. свойство [БинарныйРежим](#)).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ПисатьMifareP (WriteMifareP)**Синтаксис:****ПисатьMifareP(**

```
[in] <ИДУстройства>,
[in] <Сектор>,
[in] <Блок>,
[out] <Данные>,
[in] <КоличествоБлоков> = 1,
[in] <КлючиСчитывателя> = 0,
[in] <КлючБ> = False,
[in] <ПропуститьПрицепы> = False,
[in] <ОткрытыйТекст> = True,
[in] <Тайм-аут> = -1,
[out] <ПозицияКлюча> = Null)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сектор>

Тип: Число. Номер первого сектора (1К: от 0..15; 4К: от 0 до 39, см. структуру памяти [Mifare 1К](#) и [Mifare 4К](#));

<Блок>

Тип: Число. Номер первого блока;

<Данные>

Тип: ДвоичныеДанные. Данные для записи (каждый блок по 16 байт);

<КоличествоБлоков>

Тип: Число. Количество блоков:

Тип карты	Секторов	Всего блоков	Всего блоков без прицепов
Mifare 1К	16	64	48
Mifare 2К	32	128	96
Mifare 4К	40	256	216

<КлючиСчитывателя>

Тип: Число. Маска авторизации по ключам в памяти считывателя. Если = 0, то используется явный ключ авторизации, который устанавливается в свойстве [КлючАвторизацииMP \(AuthorizationKeyMP\)](#);

<КлючБ>

Тип: Булево. Авторизоваться по ключу Б (иначе – по ключу А);

<ПропуститьПрицепы>

Тип: Булево. Пропустить блоки-прицепы. В блоке-прицепе хранятся настройки доступа к сектору: ключ авторизации А и Б, биты доступа. Если в блок-прицеп записать некорректное значение – весь сектор становится недоступным навсегда;

<ОткрытыйТекст>

Тип: Булево. Истина (True), открытый текст, иначе – закрытый текст;

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут записаны блоки или пока не произойдет ошибка;

<ПозицияКлюча>

Тип: Число. Позиция найденного ключа авторизации при авторизации по маске ключей считывателя. Если = -1, то не найден.

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.

Константа	Значение	Описание
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_AUTHORIZE	-2147220965 (0x8004021B)	Ошибка авторизации ключа.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем записаны блоки, то функция вернёт E_PENDING и запись блоков продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить запись используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилась ли запись используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#).

Описание:

Записывает один или несколько блоков в [Mifare Plus SL3](#).

Поддерживает считыватель [Z-2 MF-I](#) в бинарном режиме (см. свойство [БинарныйРежим \(BinaryMode\)](#)).

Примечание

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader)

Синтаксис:

ПисатьКлючиMifareВСчитыватель(

```
[in] <ИДУстройства>,
[in] <Адрес>,
[in] <Ключи>,
[in] <ДлинныйКлюч>,
[in] <Тайм-аут> = -1)
```

Параметры:

<ИДУстройства>

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Адрес>

Тип: Число. Начальная позиция записи: 0..15 ключи А, 16..31 ключи Б;

<Ключи>

Тип: ДвоичныеДанные, Строка. Ключи авторизации. Если тип ="Строка", то это список ключей в 16-ричном формате, разделённых символом ';' (например, "FFFFFFFFFFFF", где младший байт слева, или "0xFFFFFFFFFFFF", где младший байт справа). Если размер ключа = 0, то ключ не записывается;

<ДлинныйКлюч>

Тип: Булево. Истина (True), размер ключа 16 байт (для [Mifare Plus](#)), иначе – 6 байт (для [Mifare Classic](#)). Размер ключа = 16 поддерживает только считыватель [Z-2 MF-I](#);

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будут записаны ключи или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.

Если тайм-аут закончился раньше, чем записаны ключи, то функция вернёт E_PENDING и запись продолжится в фоновом режиме, уведомление о завершении придёт через ВнешнееСобытие(ID устройства, [Команда](#), ""). Чтобы отменить запись используйте функцию [ПрерватьКоманду \(BreakCommand\)](#). Чтобы проверить завершилась ли запись используйте функцию [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#).

Описание:

Записывает один или несколько ключей авторизации Mifare в память считывателя. Поддерживают считыватели [Z-2 USB MF](#), [Z-2 MF-I](#), [Matrix III Net](#), [CP-Z2-MF](#) в бинарном режиме (см. свойство [БинарныйРежим \(BinaryMode\)](#)).

Примечание

У считыватель [Z-2 MF-I](#) есть 2 списка: 1) для ключей Mifare Classic, по 6 байт, 2) для ключей Mifare Plus SL3, по 16 байт.

Метод Сигнализировать (Signal)**Синтаксис:**

```
Сигнализировать(
[in] <ИДУстройства>,
[in] <Сигналы> = Null,
[in] <Сброс> = True)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Сигналы>

Тип: Строка. Строка с параметрами сигнализирования, состоящая из списка "состояний", разделенных символом ';' (точка с запятой). Каждое "состояние" состоит из параметров, разделенных симполом ' ' (пробел). Например, "1 2 3 500; 1 2 3 500", где "1" – код состояния красного светодиода, "2" – состояние зеленого светодиода, "3" – состояние зумера (звук), "500" – продолжительность состояния в миллисекундах (в дианазоне 0 – 15937,5 с дискретностью 62,5). Максимальное количество "состояний" в очереди = 15.

Код состояния	Описание
0	Не изменять
1	Включить
2	Выключить

<Сброс>

Тип: Булево. Истина (True), чтобы начать новое сигнализирование (очистить текущую очередь "состояний"), иначе продолжить текущее (добавить в очередь).

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.

Описание:

Сигнализирует с помощью индикаторов и звука считывателя. Поддерживают считыватели [Z-2 USB MF](#), [Z-2 MF-I](#), [Matrix III Net](#), [CP-Z2-MF](#) в бинарном режиме (см. свойство [БинарныйРежим \(BinaryMode\)](#)).

Примечание

После завершения сигнализирования ZR1C установит авто состояние индикаторов (если карта в поле, то включит зелёный светодиод, иначе – красный светодиод).

Очередь "состояний" можно периодически пополнять, но нельзя превысить размер очереди =15.

Метод ИнициализироватьTemic (InitializeTemic)**Синтаксис:**

```
ИнициализироватьTemic(
  [in] <ИДУстройства>,
  [in] <Тайм-аут> = -1)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будет инициализирован Temic или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .

Константа	Значение	Описание
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.

Описание:

Записывает стандартную конфигурацию в [Temic](#) (0x40801400), которая позволяет читать номер Temic с параметрами сканирования: тип = 2, скорость = 0. Поддерживают считыватели [Z-2 RD_ALL](#), [Z-2 EHR](#).

Примечание

Команда предназначена для подготовки новой карты, когда она не обнаруживается при поднесении.

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод Искать Temic (FindTemic)**Синтаксис:**

```

BOOL FindTemic(
    [in] <ИДУстройства>,
    [in] <ДлинныйКод> = False,
    [out] <НомерКлюча>,
    [in, out] <ПараметрСканирования> = -1,
    [in] <Пароль> = "",
    [in] <Тайм-аут> = -1)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<ДлинныйКод>

Тип: Булево. Истина (True), читать весь код производителя, иначе – только UID.

<НомерКлюча>

Тип: ДвоичныеДанные. Считанный номер Temic (весь код производителя: 8 байт, только UID: 5 байт);

<ПараметрСканирования>

Тип: Число. Параметр сканирования. Если = -1, то авто выбор, при этом используются параметры из свойства [ПараметрыСканированияTemic \(TemicScanParameters\)](#);

<Пароль>

Тип: Число, Строка, ДвоичныеДанные. Пароль для доступа к данным Temic. Если = "", то нет пароля. Если тип = "Строка", то строка в 16-ричном формате (например, "A1B2C3D4", где младший байт слева, или "0xD4C3B2A1", где младший байт справа). По умолчанию = "";

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будет считан номер Temic или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.

Константа	Значение	Описание
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Другая ошибка считывателя.

Описание:

Ищет карту [Temic](#). Поддерживают считыватели [Z-2 RD_ALL](#), [Z-2 EHR](#).

Примечание

Карта (новая чистая карта) может не находится без предварительной инициализации функцией [Инициализировать Temic \(InitializeTemic\)](#).

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод Читать Temic (ReadTemic)**Синтаксис:**

```

ЧитатьTemic(
  [in] <ИДУстройства>,
  [in] <Адрес>,
  [out] <Данные>,
  [in] <КоличествоБлоков> = 1,
  [in,out] <ПараметрСканирования> = -1,
  [in] <Пароль> = "",
  [in] <Тайм-аут> = -1)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Адрес>

Тип: Число. Номер блока [Temic](#) (0..7);

<Данные>

Тип: ДвоичныеДанные. Считанные данные;

<КоличествоБлоков>

Тип: Число. Количество блоков Temic;

<ПараметрСканирования>

Тип: Число. Параметр сканирования. Если = -1, то не определён. Младший байт содержит тип, старший байт – скорость (см. [параметры сканирования](#));

<Пароль>

Тип: Число, Строка, ДвоичныеДанные. Пароль для доступа к данным Temic. Если = "", то нет пароля. Если тип = "Строка", то строка в 16-ричном формате (например, "A1B2C3D4", где младший байт слева, или "0xD4C3B2A1", где младший байт справа);

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будет считан номер Temic или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_RCVTIMEOUT	-2147220967 (0x80040219)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220969 (0x80040217)	Считыватель прислал некорректный ответ.
ZR1C_E_READERERROR	-2147220968 (0x80040218)	Другая ошибка считывателя.

Описание:

Читает данные из блоков карты [Temic](#). Поддерживают считыватели [Z-2 RD_ALL](#), [Z-2 EHR](#).

Примечание

Карта (новая чистая карта) может не находится без предварительной инициализации функцией [ИнициализироватьTemic \(InitializeTemic\)](#).

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ПисатьTemic (WriteTemic)**Синтаксис:**

```

ПисатьTemic(
[in] <ИДУстройства>,
[in] <Адрес>,
[in] <Данные>,
[in] <КоличествоБлоков> = 1,
[in,out] <ПараметрСканирования> = -1,
[in] <Пароль> = "",
[in] <ЗапретитьПерезапись> = False,
[in] <Тайм-аут> = -1)

```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Адрес>

Тип: Число. Номер блока [Temic](#) (0..7);

<Данные>

Тип: Число, Строка, ДвоичныеДанные. Записываемые данные. Если тип = "Строка", то байты в 16-ричном формате (например, "A1B2C3D4", где младший байт слева, или "0xD4C3B2A1", где младший байт справа);

<КоличествоБлоков>

Тип: Число. Количество блоков Temic;

<ПараметрСканирования>

Тип: Число. Параметр сканирования. Если = -1, то не определён. Младший байт содержит тип, старший байт – скорость (см. [параметры сканирования](#));

<Пароль>

Тип: Число, Строка, ДвоичныеДанные. Пароль для доступа к данным Temic. Если = "", то нет пароля. Строка в 16-ричном формате (например, "A1B2C3D4", где младший байт слева, или "0xD4C3B2A1", где младший байт справа);

<ЗапретитьПерезапись>

Тип: Булево. Истина (True), запретить дальнейшую перезапись блока (сделать доступным только для чтения);

<Тайм-аут>

Тип: Число. Тайм-аут ожидания. Если = -1, то функция не завершится пока не будет считан номер Temic или пока не произойдет ошибка;

Возвращаемое значение:

Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_PENDING	-2147483638 (0x8000000A)	Команда в процессе выполнения.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией Подключить (Open) .
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается считывателем.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет карты.
ZR1C_E_RCVTIMEOUT	-2147220967 (0x80040219)	Тайм-аут получения ответа от считывателя.
ZR1C_E_BADANSWER	-2147220969 (0x80040217)	Считыватель прислал некорректный ответ.
ZR1C_E_READERERROR	-2147220968 (0x80040218)	Другая ошибка считывателя.

Описание:

Пишет данные в блоки карты [Temic](#). Поддерживают считыватели [Z-2 RD_ALL](#), [Z-2 EHR](#).

Примечание

Блоки пишутся в обратном порядке с 7 по 0. Если используется пароль и пишется новый пароль в блок 7, то после успешной записи блока 7 далее используется новый пароль.

Этот метод приостанавливает сканирование на время, установленное в свойстве [ТаймаутУдерживанияКлюча \(KeyHoldTimeout\)](#).

Метод ОчиститьЛог (ClearLog)**Синтаксис:**

```
ОчиститьЛог(
  [in] <УдалитьФайл> = False
)
```

Параметры:**<УдалитьФайл>**

Тип: Булево. Истина (True), удалить файл, иначе – очистить содержимое файла.

Возвращаемое значение:Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
ZR1C_E_FILEACCESS	-2147220987 (0x80040205)	Нет доступа к файлу.

Описание:

Очищает лог файл "%AppData%\IronLogic\ZR1C\ZR1C.log".

Метод ДобавитьВЛог (AddLog)**Синтаксис:**

```
ДобавитьВЛог(
  [in] <Тип> = 0,
  [in] <ID> = 0,
  [in] <Тэги> = "",
  [in] <Текст>
)
```

Параметры:**<Тип>**

Тип: Число. Тип сообщения:

Значение	Описание
0	Информация
1	Предупреждение
2	Ошибка

<ID>

Тип: Число. ID сообщения;

<Тэги>

Тип: Строка. Тэги;

<Текст>

Тип: Строка. Текст сообщения.

Возвращаемое значение:Тип: Булево. Истина (True), команда выполнена успешно, иначе – ошибка, информацию об ошибке можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Константа	Значение	Описание
E_INVALIDARG	-2147024809 (0x80070057)	Указаны неправильные параметры.
E_ABORT	-2147467260 (0x80004004)	Команда прервана. Запись лог файла выключена (свойство LogLevel = 0).
E_OUTOFMEMORY	-2147024882 (0x8007000E)	Недостаточно памяти для обработки команды.
ZR1C_E_FILEACCESS	-2147220987 (0x80040205)	Нет доступа к файлу.
ZR1C_E_DISK_FULL	-2147220986 (0x80040206)	Нет места на жёстком диске.

Описание:

Добавляет сообщение в лог файл "%AppData%\IronLogic\ZR1C\ZR1C.log".

События компоненты

Сообщения о событиях компоненты приходят в IC в обработке события ВнешнееСобытие (ExternalEvent).

Событие	Описание
События для поддержки технологии "IC совместимо"	
ДанныеКарты (TracksData)	Уведомляет о подносе карты в поле считывателя.
ОшибкаДрайвера (DriverError)	Уведомляет об ошибке при работе со считывателем.
Дополнительные	
УстройствоПодключено (DeviceConnect)	Уведомляет об успешном подключении к считывателю.
УстройствоОтключено (DeviceDisconnect)	Уведомляет о потере связи со считывателем.
НетКарты (NoCard)	Уведомляет об удалении карты из поля считывателя.
Комманда (Command)	Уведомляет о завершении фоновой команды.

Дополнительные события по умолчанию отключены, их можно включить в программе "Конфигуратор ZR1C" на странице «[Основное](#)».

Событие ДанныеКарты (TracksData)**Синтаксис:**

```
ВнешнееСобытие(
  [in] <ИДУстройства>,
  [in] <Событие>,
  [in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = ДанныеКарты (TracksData);

<Данные>

Тип: Строка. Форматированный номер поднесенной карты.

Описание:

Уведомляет о подносе карты в поле считывателя.

Примечание:

Форматированный номер текущей карты в поле считывателя также можно получить методом [GetKeyInfo](#).

Событие ОшибкаДрайвера (DriverError)**Синтаксис:**

```
ВнешнееСобытие(  
  [in] <ИДУстройства>,  
  [in] <Событие>,  
  [in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = ОшибкаДрайвера (DriverError);

<Данные>

Тип: Строка. Текст с описанием ошибки.

Описание:

Уведомляет об ошибке при работе со считывателем.

Событие УстройствоПодключено (DeviceConnect)**Синтаксис:**

```
ВнешнееСобытие(  
  [in] <ИДУстройства>,  
  [in] <Событие>,  
  [in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = УстройствоПодключено (DeviceConnect);

<Данные>

Тип: Строка. Пустая строка.

Описание:

Уведомляет об успешном подключении к считывателю.

Событие УстройствоОтключено (DeviceDisconnect)**Синтаксис:**

```
ВнешнееСобытие(  
[in] <ИДУстройства>,  
[in] <Событие>,  
[in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = УстройствоОтключено (DeviceDisconnect);

<Данные>

Тип: Строка. Пустая строка.

Описание:

Уведомляет о потере связи со считывателем.

Событие НетКарты (NoCard)**Синтаксис:**

```
ВнешнееСобытие(  
[in] <ИДУстройства>,  
[in] <Событие>,  
[in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = НетКарты (NoCard);

<Данные>

Тип: Строка. Форматированный номер удаленной карты.

Описание:

Уведомляет об удалении карты из поля считывателя.

Событие Комманда (Command)**Синтаксис:**

```
ВнешнееСобытие(  
[in] <ИДУстройства>,  
[in] <Событие>,  
[in] <Данные>)
```

Параметры:**<ИДУстройства>**

Тип: Строка. Идентификатор устройства (имя порта считывателя);

<Событие>

Тип: Строка. Название события = Комманда (Command);

<Данные>

Тип: Строка. Пустая строка.

Описание:

Уведомляет о завершении фоновой команды.

Примечание:

Возможные фоновые команды:

Метод	Описание
ЧитатьMifareUL (ReadMifareUL)	Читает данные карты Mifare Ultralight .
ПисатьMifareUL (WriteMifareUL)	Пишет данные на карту Mifare Ultralight .
ЧитатьMifareC (ReadMifareC)	Читает данные карты Mifare Classic .
ПисатьMifareC (WriteMifareC)	Пишет данные на карту Mifare Classic .
ЧитатьMifareP (ReadMifareP)	Читает данные карты Mifare Plus SL3 .
ПисатьMifareP (WriteMifareP)	Пишет данные на карту Mifare Plus SL3 .
ПисатьКлючиMifareВСчитыватель (WriteMifareKeysToReader)	Записывает один или несколько ключей авторизации Mifare в память считывателя.
Сигнализировать (Signal)	Сигнализирует с помощью индикаторов и звука считывателя.
ИнициализироватьTemic (InitializeTemic)	Записывает стандартную конфигурацию в Temic .
ИскатьTemic (FindTemic)	Ищет карту Temic .
ЧитатьTemic (ReadTemic)	Читает данные из блоков карты Temic .
ПисатьTemic (WriteTemic)	Ищет данные в блоки карты Temic .

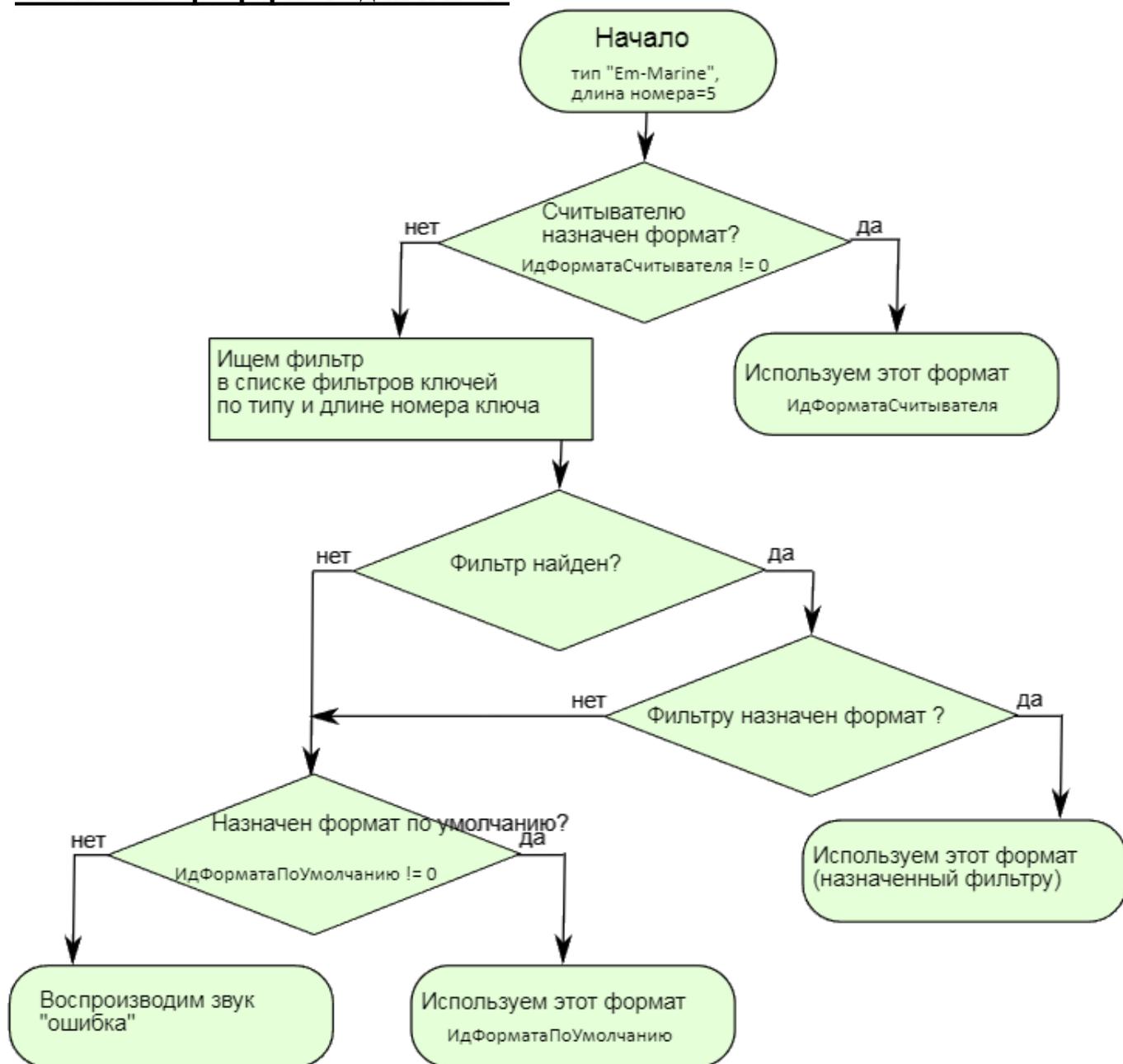
Команду можно прервать методом [ПрерватьКоманду \(BreakCommand\)](#).

Получить состояние команды можно методом [ПолучитьСостояниеКоманды \(GetCommandStatus\)](#).

Получить результат выполнения команды (данные) можно методом [ПолучитьРезультатКоманды \(GetCommandResult\)](#).

Схема передачи номера ключа

Схема выбора формата для ключа



Коды возврата ZR1C

Константа	Значение	Описание
S_OK	0	Команда выполнена успешно.
E_PENDING	-2147483638 (0x8000000A)	Команда выполняется в фоновом режиме.
E_BOUNDS	-2147483637 (0x8000000B)	Индекс вне диапазона.
E_ABORT	-2147467260 (0x80004004)	Команда прервана.
E_FAIL	-2147467259 (0x80004005)	Неизвестная ошибка.
E_OUTOFMEMORY	-2147024882 (0x8007000E)	Недостаточно памяти для обработки команды.
E_INVALIDARG	-2147024809 (0x80070057)	Неправильные параметры.

Константа	Значение	Описание
ZR1C_E_PARAMNOTEXIST	-2147220991 (0x80040201)	Параметр не существует.
ZR1C_E_INVALIDPARAMTYPE	-2147220990 (0x80040202)	Неправильный тип параметра.
ZR1C_E_OUT_OF_RESOURCES	-2147220989 (0x80040203)	Недостаточно ресурсов для завершения операции. Когда не удалось создать событие, поток и т.п.
ZR1C_E_WSASTARTUP_FAILED	-2147220988 (0x80040204)	Не удалось запустить Winsock.
ZR1C_E_FILEACCESS	-2147220987 (0x80040205)	Нет доступа к файлу.
ZR1C_E_DISK_FULL	-2147220986 (0x80040206)	Нет места на жёстком диске.
ZR1C_E_INVALIDPORTNAME	-2147220976 (0x80040210)	Некорректное имя порта.
ZR1C_E_PORTNOTEXIST	-2147220975 (0x80040211)	Порт не существует.
ZR1C_E_PORTBUSY	-2147220974 (0x80040212)	Порт занят.
ZR1C_E_PORTDISCONNECTED	-2147220973 (0x80040213)	Порт отключен.
ZR1C_E_CONNECTTIMEOUT	-2147220971 (0x80040215)	Тайм-аут подключения по TCP.
ZR1C_E_SENDTIMEOUT	-2147220970 (0x80040216)	Тайм-аут отправки данных запроса в порт.
ZR1C_E_RCVTIMEOUT	-2147220969 (0x80040217)	Тайм-аут получения данных ответа из порта.
ZR1C_E_BADANSWER	-2147220968 (0x80040218)	Считыватель прислал некорректный ответ.
ZR1C_E_READERERROR	-2147220967 (0x80040219)	Считыватель вернул ошибку.
ZR1C_E_NOKEY	-2147220966 (0x8004021A)	Нет ключа.
ZR1C_E_AUTHORIZE	-2147220965 (0x8004021B)	Ошибка авторизации ключа.
ZR1C_E_KEYACCESS	-2147220964 (0x8004021C)	Ошибка доступа к данным ключа.
ZR1C_E_KEYWRITELOCK	-2147220963 (0x8004021D)	Ошибка авторизации ключа.
ZR1C_E_CMDNOTSUPPORTED	-2147220962 (0x8004021E)	Команда не поддерживается.
ZR1C_E_UNKNOWNMESSAGE	-2147220961 (0x8004021F)	Не распознано сообщение считывателя. Вероятно не заводская прошивка.
ZR1C_E_FINDTIMEOUT	-2147220944 (0x80040230)	Тайм-аут поиска считывателей.
ZR1C_E_NOMOREREADERS	-2147220943 (0x80040231)	Нет больше считывателей.
ZR1C_E_READERNOTFOUND	-2147220942 (0x80040232)	Считыватель не найден.
ZR1C_E_NOTREGISTERED	-2147220941 (0x80040233)	Считыватель не зарегистрирован функцией "Подключить".

Константа	Значение	Описание
ZR1C_E_INVALIDFORMATNAME	-2147220940 (0x80040234)	Некорректное название формата.
ZR1C_E_FORMATNAMEEXIST	-2147220939 (0x80040235)	Формат с указанным именем уже существует.
ZR1C_E_NOTENOUGHPARAMS	-2147220938 (0x80040236)	Недостаточно параметров.
ZR1C_E_SPECNOTSUPPORTED	-2147220937 (0x80040237)	Спецификатор шаблона не поддерживается.
ZR1C_E_UNKNOWNKEYWORD	-2147220936 (0x80040238)	Неизвестное ключевое слово.
ZR1C_E_ENDEXPRESSION	-2147220935 (0x80040239)	Неожиданный конец выражения.
ZR1C_E_UNKNOWNSYMBOL	-2147220934 (0x8004023A)	Неизвестный символ.
ZR1C_E_LBRACKET	-2147220933 (0x8004023B)	Ожидается '(' или '['.
ZR1C_E_RBRACKET	-2147220932 (0x8004023C)	Ожидается ')' или ']'.
ZR1C_E_NUMBERSYNTAX	-2147220931 (0x8004023D)	Ошибка синтаксиса числа.
ZR1C_E_UNEXPECTED	-2147220930 (0x8004023E)	Неожиданный оператор.
ZR1C_E_INVALIDKEYFILTERNAME	-2147220928 (0x80040240)	Некорректное название фильтра ключей.
ZR1C_E_KEYFILTERNAMEEXIST	-2147220927 (0x80040241)	Фильтр ключей с указанным названием уже существует.
ZR1C_E_KEYFILTERLIMIT	-2147220926 (0x80040242)	Достигнут предел количества фильтров ключей, максимум 31 (0x1F).
ZR1C_E_FORMATLIMIT	-2147220925 (0x80040243)	Достигнут предел количества форматов, максимум 4294967295 (0xffffffff).

Если [метод](#) возвращает Ложь (False), значит произошла ошибка, код и описание которой можно получить методом [ПолучитьОшибку \(GetLastError\)](#).

Формат Шаблон

Значение поля "*Шаблон*" состоит из обычных символов, специальных *управляющих последовательностей* символов и *спецификаторов формата*.

Обычные символы и управляющие последовательности просто копируются в стандартный выходной поток в порядке их появления.

Спецификатор определяется символом "%", за которым может следовать до пяти полей в следующем порядке:

%[*флаги*] [*ширина*] [*точность*] [*размер*] *тип*

Количество спецификаторов формата должно быть равно количеству выводимых значений, которые указываются в поле "*Параметры*".

В квадратных скобках указаны необязательные поля.

тип

Обязательное поле *тип* задаёт тип выводимого значения.

Символ	Тип	Формат вывода
d	int	Знаковое десятичное целое.
i	int	Знаковое десятичное целое.
o	int	Беззнаковое восьмеричное целое.
u	int	Беззнаковое десятичное целое.
x	int	Беззнаковое шестнадцатеричное целое с использованием символов «abcdef».
X	int	Беззнаковое шестнадцатеричное целое с использованием символов «ABCDEF».

флаги

Флаг	Действие	По умолчанию
-	Выравнивание по левому краю в пределах заданной ширины.	Выравнивание по правому краю.
+	Добавление знака "+" или "-" перед числами.	Знак добавляется только перед отрицательными числами.
0	Добавление нулей перед выводимым значением. Если одновременно используются флаги "-" и "0", то "0" игнорируется.	Добавление пробелов.
пробел	Добавление пробела перед положительным числом. Если одновременно используются флаги "пробел" и "+", то "пробел" игнорируется.	Пробел не добавляется.
#	Добавление символов 0, 0x или 0X перед ненулевым значением, если флаг # используется с форматами o, x или X соответственно.	Символы 0, 0x и 0X не добавляются.
	При использовании с форматами e, E, f и F флага # выводимое число будет содержать десятичную точку в любом случае.	Десятичная точка добавляется, только если за ней следуют цифры.
	При использовании с форматами g и G флага # выводимое число будет содержать десятичную точку и хвостовые нули в любом случае.	Десятичная точка добавляется, только если за ней следуют цифры. Хвостовые нули не выводятся.
	Игнорируется при использовании с форматами c, d, i, u и s.	

ширина

Поле *ширина* содержит минимальное количество выводимых символов – неотрицательное целое число. Если выводимое значение содержит меньше символов, то оно расширяется пробелами (если не задан флаг 0). Если поле *ширина* содержит звездочку (*), то в качестве значения поля берётся целое число из списка аргументов, предшествующее выводимому значению.

точность

Поле *точность* также представляет собой неотрицательное целое число. Действие зависит от *типа* выводимого значения.

Тип	Действие	По умолчанию
d, i, u, o, x, X	Точность задаёт минимальное количество символов, которые будут напечатаны. Если число содержит меньше символов, оно расширяется нулями.	Точность равна 1.

Если поле *точность* содержит звездочку (*), то в качестве значения поля берётся целое число из списка аргументов, предшествующее выводимому значению.

размер

Поле *размер* позволяет указать размер данных, переданных шаблону. Набор применяемых спецификаторов размера зависит от спецификатора типа.

Спецификатор	%d, %i, %o, %u, %x, %X	%s, %C
отсутствует	8 байт	2 байта для %s, 1 байт для %C
h	2 байта	1 байт
l	4 байта	2 байта
ll	8 байт	-
w	-	2 байта
I	4 байта в 32-битной версии ZR1C , 8 байт – в 64-битной	-
I32	4 байта	-
I64	8 байт	-

К управляющим последовательностям относятся следующие последовательности символов.

Последовательность	Действие
\n	Новая строка
\r	Возврат каретки (соответствует нажатию клавиши Enter, аналогично \15 или \#13 или \x0d)
\t	Горизонтальная табуляция
\'	Одиночная кавычка (апостроф)
\"	Двойная кавычка
\\	Обратная косая черта
\?	Литерал вопросительного знака
\ooo	ASCII символ в восьмеричной нотации
\xhhhh	ASCII символ в шестнадцатеричной нотации
\Shhhh	ASCII символ в шестнадцатеричной нотации
\#ddd	ASCII символ в десятичной нотации

Параметры

Значение поля "*Параметры*" состоит из параметров, соответствующих спецификаторам в шаблоне.

Параметры разделяются пробелами. Каждый параметр может состоять из математических выражений, целых чисел в 8-, 10- и 16-ричном виде.

Синтаксис целых чисел:

[*разрядность*]*цифры*

0x – 16-ричное число, цифры: 0 1 2 3 4 5 6 7 8 9 a b c d e f. Регистр букв в префиксе и в цифрах не учитывается.

0b – 2-ичное число, цифры: 0 1. Регистр букв в префиксе и в цифрах не учитывается.

0 – 8-ричное число, цифры: 0 1 2 3 4 5 6 7

по умолчанию (без префикса) – 10-тичное число, цифры: 0 1 2 3 4 5 6 7 8 9

Константы:

Константа	Размер, байт	Описание
x[A]	1	Значение байта с индексом А в номере ключа
bA	1	Байт – целое число с диапазоном значений 0..255 (беззнаковое) или -128..+127 (знаковое). Если А – цифра в 10-тичном виде, то = x[A]. Если А = 'Z', то = 0; Если А = 'X', то = ds(1, x[0], x[1], x[2], x[3], x[4], x[5]); Если А = 'Y', то = ds(1, x[0], x[1], x[2], 0, 0, 0); Если А = 'L', то = длина номера в байтах.
wAB	2	Слово – целое число с диапазоном значений 0..65535 или -32768...+32767. = bA (bB << 8)
lABCD	4	Двойное слово – целое число с диапазоном значений 0..4294967295 или -2147483648..+2147483647. = bA (bB << 8) (bC <<16) (bD << 24)
iABCD	4	идентично lABCD
m01234567	8	64-битное целое число с диапазоном значений 0..18446744073709551615 или -9223372036854775808..+9223372036854775807. = b0 (b1 << 8) (b2 <<16) (b3 << 24) (b4 << 32) (b5 << 40) (b6 << 48) (b7 << 54)
hA	1/2	Полубайт – целое число с диапазоном значений 0..15 или -8..+7. Где А – индекс полубайта. =(x[A/2] >> (4 * (A & 1))) & 0xF
bhAB	1	Байт. Где А,В – индексы полубайтов.
whABCD	2	Слово. Где А,В,С,Д – индексы полубайтов.
lh01234567	4	Двойное слово. Где 0,...,7 – индексы полубайтов.
mh0123456789ABCDEF	8	64-битное целое. Где 0,...,F – индексы полубайтов.

Операторы:

Оператор	Приоритет	Выполняемая операция
Арифметические		
a + b	4	Сложение
a - b	4	Вычитание
+a	2	
-a	2	Изменение знака
a * b	3	Умножение
a / b	3	Деление
a % b	3	Остаток от деления
a ^ b	7	Поразрядное исключающее ИЛИ
Логические		
a & b	6	Поразрядное логическое И
a && b	9	Логическое И (конъюнкция)
a b	8	Поразрядное логическое ИЛИ
a b	10	Логическое ИЛИ (дизъюнкция)
!a	2	Логическое отрицание (НЕ) (отрицание)
~a	2	Побитовое отрицание (НЕ)

$a \ll b$	5	Логический сдвиг влево
$a \gg b$	5	Логический сдвиг вправо
Другие		
()	1	Круглые скобки
[]	1	Операция индексирования

Операторы в порядке уменьшения приоритета: (,); -, !, ~; *, /, %; +, -; <<, >>; &; ^; |; &&; ||.

Функции:

Функция	Описание
$a = ds(\text{size}, b, c, d, \dots)$	Вычисление контрольной суммы Dallas. Параметры: size – размер следующих аргументов в байтах, b, c, d, ... – числа от младшего к старшему.
$a = rbit(\text{size}, b)$	Переворачивание бит. Параметры: size – размер следующего аргумента в байтах, b – число
$a = cbit(\text{size}, b)$	Вычисляет количество установленных бит. Параметры: size – размер следующего аргумента в байтах, b – число
$a = w(26, \text{size}, b, c, \dots)$	Вычисляет номер в кодировке Wiegand с добавлением признака кодировки. Параметры: 26 – задает кодировку с 26 по 64, size – размер следующих аргументов в байтах, b, c, ... – числа от младшего к старшему.
$a = w2(26, \text{size}, b, c, \dots)$	Вычисляет номер в кодировке Wiegand без добавления признака кодировки. Параметры: 26 – задает кодировку с 26 по 64, size – размер следующих аргументов в байтах, b, c, ... – числа от младшего к старшему.
$a = set(\text{idx}, a)$	Сохраняет значение a во временную переменную с номером idx , затем возвращает значение a .
$a = get(\text{idx})$	Возвращает значение a , которое ранее было сохранено во временную переменную с номером idx .
$a = \log_{10}(b)$	Вычисляет десятичный логарифм. Если $b = 0$, то возвращает -1.

Ключи

- [Mifare Classic 1K](#)
- [Mifare Classic 4K](#)
- [Mifare Plus](#)
- [Mifare Ultralight](#)
- [Temic \(T5557, T5577\)](#)

Mifare Classic 1K

Phillips разработал карты Mifare 1K для использования со считывателями стандарта ISO/IEC14443A. Радиointерфейс (MIFARE RF) соответствует частям 2 и 3 стандарта ISO/IEC14443A. Слой безопасности поддерживает доказанный для поля шифр потока CRYPTO1 для безопасного обмена данными mifare классического семейства.

Основные возможности Mifare Classic 1K

1. Радиочастотный интерфейс Mifare (ISO/IEC 14443A)
 - Бесконтактная передача данных и питание по радиоканалу (не требует батарей)
 - Рабочее расстояние до 100мм (зависит от геометрических параметров антенны)
 - Рабочая частота: 13,56МГц
 - Быстрая передача данных: 106Кбит/сек
 - Высокая надежность передачи (16-битовая CRC, проверка на четность...)
 - Настоящая антиколлизия (поддержка нескольких карт в поле одновременно)
 - Время типовой транзакции менее 100мс (в том числе резервное управление)

2. Память

- Размер 1КБайт, организована в виде 16 секторов, каждый из которых разбит на 4 блока, по 16 байт каждый
- Определяемые пользователем условия доступа для каждого блока памяти
- Срок хранения данных – 10 лет
- 100 000 циклов записи

3. Защита

Настоящая трехпроходная аутентификация (ISO/IEC DIS9798-2)

- Шифрование данных в радиоканале с защитой от повтора нападения
- Индивидуальная пара ключей для каждого сектора
- Уникальный 4-байтовый серийный номер для каждой карты
- Транспортный ключ защищает доступ к EEPROM при поставке чипа

Специальный акцент был сделан на безопасности против подделок. Настоящая двусторонняя идентификация и шифрование данных сообщения защищают карту от любого вида вмешательства и поэтому делают ее привлекательной в качестве билетов. Серийные номера не могут изменяться, что гарантирует уникальность каждой карты.

Организация памяти Mifare Classic 1K

1024x8 бит перепрограммируемой памяти организовано в виде 16 секторов с 4 блоками в каждом по 16 байт в каждом блоке.

В стертом состоянии ячейки читаются, как логический ноль, в записанном – как 1.

Сектор	Блок	Номер байта в блоке														Описание
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	
15	3	Key A				Access Bits				Key B						Sector Trailer 15
	2	Data														Data
	1	Data														Data
	0	Data														Data
14	3	Key A				Access Bits				Key B						Sector Trailer 14
	2	Data														Data
	1	Data														Data
	0	Data														Data
:	:															
:	:															
:	:															
1	3	Key A				Access Bits				Key B						Sector Trailer 1
	2	Data														Data
	1	Data														Data
	0	Data														Data
0	3	Key A				Access Bits				Key B						Sector Trailer 0
	2	Data														Data
	1	Data														Data
	0	Data														Manufacturer Block

1. Блок производителя (Manufacturer Block)

Нулевой блок хранит данные производителя. Уникальный (гарантировано Philips) ID, или серийный номер карты – байт 0..3. Четвертый байт – контрольная сумма номера. Блок данных производителя доступен только для чтения.

2. Блоки данных (Data)

Сектора 0..15 содержат по 3 блока данных. Блоки данных могут быть сконфигурированы с помощью битов доступа (access bits) для чтения-записи или для хранения значения (value).

Блоки значения (value)

Блоки значения позволяют выполнять команды read (чтение), write(запись), increment (увеличение), decrement (уменьшение), restore (восстановление) и transfer (сохранение). Блок значения имеет фиксированный формат, позволяющий обнаружение и исправление ошибок. Блок значения может быть сгенерирован только командой записи.

Номер байта	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Описание	значение (value)				НЕ значение (not value)				значение (value)				Адр.	НЕ адр.	Адр.	НЕ адр.

Значение (value) – 4 байтовое знаковое целое число. (not value) – 4 байтовое инверсное к value знаковое целое число. Адрес (адр) – 1-байтовый адрес, который может быть использован для реализации функции бэкапа. Изменяется только командой записи.

3. Прицеп сектора (Sector trailer)

Каждый сектор имеет "прицеп", расположенный в блоке №3 каждого сектора. Каждый прицеп хранит секретные ключи А и Б, условия доступа для всех блоков в секторе (байты 6..9).

Номер байта	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Описание	Ключ А (key A)						Биты доступа			Ключ Б (key B)						

Работа с памятью Mifare Classic 1K

Карта состоит из 16 независимых секторов.

Каждый сектор состоит из 4 блоков (3 информационных и 1 для хранения ключей и условий доступа).

Блок является самым малым компонентом, к которому адресуется пользователь, и состоит из 16 байт.

Каждый сектор имеет свой набор ключей доступа, что позволяет разграничивать доступ к различным приложениям.

Доступ к секторам памяти может настраиваться пользователем при различных его условиях.

Условия доступа

Условия доступа к областям данных и к прицепу задаются тремя битами, которые хранятся в прицепе. Биты доступа задают, какие ключи необходимы для операций над областью.



№ области (=№ блока)	Описание	Биты доступа	Допустимые команды
0	область данных	C10 C20 C30	read, write, increment, decrement, transfer, restore
1	область данных	C11 C21 C31	read, write, increment, decrement, transfer, restore
2	область данных	C12 C22 C32	read, write, increment, decrement, transfer, restore
3	прицеп сектора	C13 C23 C33	read, write

ВНИМАНИЕ: если биты доступа записаны неверно, то сектор становится недоступен – то есть необратимо блокируется и использовать его в дальнейшем НЕ сможет никто!!!

Условия доступа для прицепа

В зависимости от значения битов доступа, прицеп сектора может быть сконфигурирован для чтения/записи, как "никогда", "ключ А", "ключ Б" или "ключ А или Б". Для новых карт, ключ А задается производителем (для Филипса: FF FF FF FF FF FF).

ВНИМАНИЕ: так как биты доступа могут блокировать доступ к самим себе, следует быть внимательным при разметке новой карты!

Биты доступа			Условия доступа для...						Примечания
			ключ А		биты доступа		ключ Б		
C1	C2	C3	чтение	запись	чтение	запись	чтение	запись	
0	0	0	никогда	ключ А	ключ А	никогда	ключ А	ключ А	ключ Б может быть прочитан
0	1	0	никогда	никогда	ключ А	никогда	ключ А	никогда	ключ Б может быть прочитан
1	0	0	никогда	ключ Б	ключ А или Б	никогда	никогда	ключ Б	
1	1	0	никогда	никогда	ключ А или Б	никогда	никогда	никогда	
0	0	1	никогда	ключ А	ключ А	ключ А	ключ А	ключ А	ключ Б может быть прочитан (новая карта)
0	1	1	никогда	ключ Б	ключ А или Б	ключ Б	никогда	ключ Б	
1	0	1	никогда	никогда	ключ А или Б	ключ Б	никогда	никогда	
1	1	1	никогда	никогда	ключ А или Б	никогда	никогда	никогда	

Условия доступа для областей данных

В зависимости от значений битов доступа, данные могут быть доступны для чтения/записи: "никогда", по ключу А, по ключу Б или по ключу А или Б. Установка битов доступа определяет допустимые команды и применение карты. блок чтения-записи: доступен и на чтение, и на запись блок значения (value): дополнительно позволяет операции increment, decrement, transfer и restore. В случае единицы (001) только чтение и декремент допустимы (для "не пополняемой" карты). В случае 110, "пополнение" возможно с помощью ключа Б. блок данных производителя всегда доступен на чтение, вне зависимости от битов доступа.

Биты доступа			Условия доступа для...				Применение
C1	C2	C3	чтение	запись	инкремент	декремент, transfer, restore	
0	0	0	ключ А или Б*	ключ А или Б*	ключ А или Б*	ключ А или Б*	новая карта*
0	1	0	ключ А или Б*	никогда	никогда	никогда	блок для чтения-записи*
1	0	0	ключ А или Б*	ключ Б*	никогда	никогда	блок для чтения-записи
1	1	0	ключ А или Б*	ключ Б*	ключ Б*	ключ А или Б*	блок значения
0	0	1	ключ А или Б*	никогда	никогда	ключ А или Б*	блок значения*
0	1	1	ключ Б*	ключ Б*	никогда	никогда	блок для чтения-записи
1	0	1	ключ Б*	никогда	никогда	никогда	блок для чтения-записи
1	1	1	никогда	никогда	никогда	никогда	блок для чтения-записи

* если ключ Б может быть прочитан (из соответствующего прицепа), он не может служить для авторизации.

При попытке авторизоваться ключом Б, карта будет отвечать отказом в любом последующем доступе.

Mifare Classic 4K

Phillips разработал карты Mifare 4K для использования со считывателями стандарта ISO/IEC14443A. Радиоинтерфейс (MIFARE RF) соответствует частям 2 и 3 стандарта ISO/IEC14443A. Слой безопасности поддерживает доказанный для поля шифр потока CRYPTO1 для безопасного обмена данными mifare классического семейства.

Основные возможности Mifare 4K Classic

1. Радиочастотный интерфейс Mifare (ISO/IEC 14443A)
 - Бесконтактная передача данных и питание по радиоканалу (не требует батарей)
 - Рабочее расстояние до 100мм (зависит от геометрических параметров антенны)
 - Рабочая частота: 13,56МГц
 - Быстрая передача данных: 106Кбит/сек
 - Высокая надежность передачи (16-битовая CRC, проверка на четность...)
 - Настоящая антиколлизия (поддержка нескольких карт в поле одновременно)
 - Время типовой транзакции менее 100мс (в том числе резервное управление)
2. Память
 - Размер 4КБайт, разбита на 32 сектора по 4 блока и на 8 секторов по 16 блоков. Один блок состоит из 16 байт.
 - Определяемые пользователем условия доступа для каждого блока памяти
 - Срок хранения данных – 10 лет
 - 100 000 циклов записи
3. Защита

Настоящие три прохода аутентификации (ISO/IEC DIS9798-2)

- Шифрование данных в радиоканале
- Индивидуальная пара ключей для каждого сектора
- Уникальный 4-байтовый серийный номер для каждой карты
- Транспортный ключ защищает доступ к EEPROM при поставке чипа

Организация памяти Mifare Classic 4K

4КБайта перепрограммируемой памяти разбиты на 32 сектора по 4 блока и на 8 секторов по 16 блоков. Один блок состоит из 16 байт.

В стертом состоянии ячейки читаются, как логический ноль, в записанном – как 1.

Сектор	Блок	Номер байта в блоке														Описание	
		0	1	2	3	4	5	6	7	8	9	10	11	12	13		14
39	15	Key A				Access Bits				Key B						Sector Trailer 39	
	14																Data
	13																Data

	2																Data
	1																Data
0																Data	
..	
..	
..	
32	15	Key A				Access Bits				Key B						Sector Trailer 32	
	14															Data	
	13															Data	
	
	2															Data	
	1															Data	
0															Data		
..		
31	3	Key A				Access Bits				Key B						Sector Trailer 31	
	2															Data	
	1															Data	
	0															Data	
..		
..		
0	3	Key A				Access Bits				Key B						Sector Trailer 0	
	2															Data	
	1															Data	
	0															Manufacturer Data	

1. Блок производителя (Manufacturer Block)

Нулевой блок хранит данные производителя. Уникальный (гарантировано Philips) ID, или серийный номер карты – байт 0..3. Четвертый байт – контрольная сумма номера. Блок данных производителя доступен только для чтения.

2. Блоки данных (Data)

Сектора 0..31 содержат по 3 блока данных, а сектора 32..39 – по 15 блоков данных. Блоки данных могут быть сконфигурированы с помощью битов доступа (access bits) для чтения-записи или для хранения значения (value).

Блоки значения (value)

Блоки значения позволяют выполнять команды read (чтение), write(запись), increment (увеличение), decrement (уменьшение), restore (восстановление) и transfer (сохранение). Блок значения имеет фиксированный формат, позволяющий обнаружение и исправление ошибок. Блок значения может быть сгенерирован только командой записи.

Номер байта	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Описание	значение (value)				НЕ значение (not value)				значение (value)				Адр.	НЕ адр.	Адр.	НЕ адр.

Значение (value) – 4 байтовое знаковое целое число. (not value) – 4 байтовое инверсное к value знаковое целое число. Адрес (адр) – 1-байтовый адрес, который может быть использован для реализации функции бэкапа. Изменяется только командой записи.

3. Прицеп сектора (Sector trailer)

Каждый сектор имеет "прицеп", расположенный в блоке №3 каждого сектора для первых 2х Килобайт и в блоке №15 каждого сектора – для старших 2х Килобайт. Каждый прицеп хранит секретные ключи А и Б, условия доступа для всех блоков в секторе (байты 6..9).

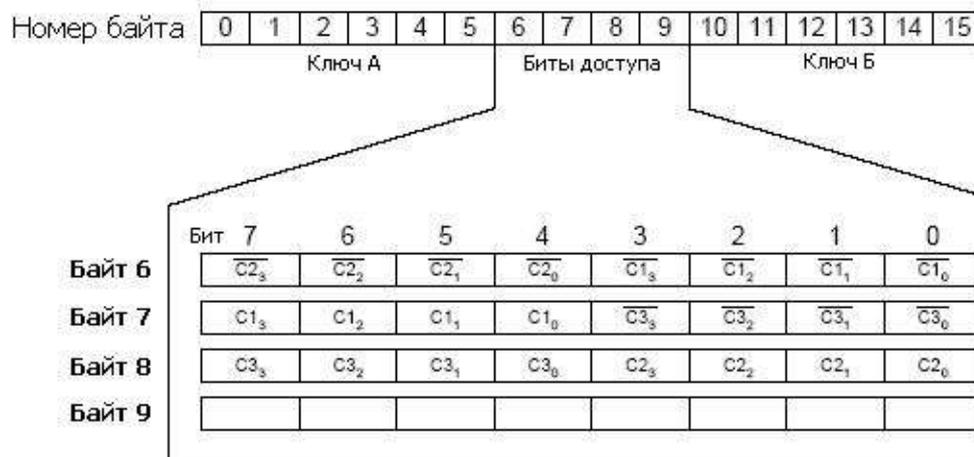
Номер байта	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Описание	Ключ А (key A)					Биты доступа				Ключ Б (key B)						

Работа с памятью Mifare Classic 4K

Условия доступа

Условия доступа к областям данных и к прицепу задаются тремя битами, которые хранятся в прицепе. Биты доступа задают, какие ключи необходимы для операций над областью.

Биты доступа	Допустимые команды			Описание
$C1_3 C2_3 C3_3$	read, write	→	3	прицеп сектора
$C1_2 C2_2 C3_2$	read, write, increment, decrement, transfer, restore	→	2	область данных
$C1_1 C2_1 C3_1$	read, write, increment, decrement, transfer, restore	→	1	область данных
$C1_0 C2_0 C3_0$	read, write, increment, decrement, transfer, restore	→	0	область данных



Примечание: для младших 2 Кбайт, область данных равна 1 блоку (16 байт). Для старших 2 Кбайт область данных = 5 блокам. Т.е. в старших 2 Кб, доступ задается не к каждому блоку индивидуально, а к областям по 5 блоков.

ВНИМАНИЕ: если биты доступа записаны неверно, то сектор становится недоступен – то есть необратимо блокируется и использовать его в дальнейшем НЕ сможет никто!!!

Условия доступа для прицепа

В зависимости от значения битов доступа, прицеп сектора может быть сконфигурирован для чтения/записи, как "никогда", "ключ А", "ключ Б" или "ключ А или Б". Для новых карт, ключ А задается производителем (для Филипса: FF FF FF FF FF FF).

ВНИМАНИЕ: так как биты доступа могут блокировать доступ к самим себе, следует быть внимательным при разметке новой карты!

Биты доступа			Условия доступа для...						Примечания
			ключ А		биты доступа		ключ Б		
C1	C2	C3	чтение	запись	чтение	запись	чтение	запись	
0	0	0	никогда	ключ А	ключ А	никогда	ключ А	ключ А	ключ Б может быть прочитан
0	1	0	никогда	никогда	ключ А	никогда	ключ А	никогда	ключ Б может быть прочитан
1	0	0	никогда	ключ Б	ключ А или Б	никогда	никогда	ключ Б	
1	1	0	никогда	никогда	ключ А или Б	никогда	никогда	никогда	
0	0	1	никогда	ключ А	ключ А	ключ А	ключ А	ключ А	ключ Б может быть прочитан (новая карта)
0	1	1	никогда	ключ Б	ключ А или Б	ключ Б	никогда	ключ Б	
1	0	1	никогда	никогда	ключ А или Б	ключ Б	никогда	никогда	
1	1	1	никогда	никогда	ключ А или Б	никогда	никогда	никогда	

Условия доступа для областей данных

В зависимости от значений битов доступа, данные могут быть доступны для чтения/записи: "никогда", по ключу А, по ключу Б или по ключу А или Б. Установка битов доступа определяет допустимые команды и применение карты. блок чтения-записи: доступен и на чтение, и на запись блок значения (value): дополнительно позволяет операции increment, decrement, transfer и restore. В случае единицы (001) только чтение и декремент допустимы (для "не пополняемой" карты). В случае 110, "пополнение" возможно с помощью ключа Б. блок данных производителя всегда доступен на чтение, вне зависимости от битов доступа.

Биты доступа			Условия доступа для...				Применение
C1	C2	C3	чтение	запись	инкремент	декремент, transfer, restore	
0	0	0	ключ А или Б*	ключ А или Б*	ключ А или Б*	ключ А или Б*	новая карта*
0	1	0	ключ А или Б*	никогда	никогда	никогда	блок для чтения-записи*
1	0	0	ключ А или Б*	ключ Б*	никогда	никогда	блок для чтения-записи
1	1	0	ключ А или Б*	ключ Б*	ключ Б*	ключ А или Б*	блок значения
0	0	1	ключ А или Б*	никогда	никогда	ключ А или Б*	блок значения*
0	1	1	ключ Б*	ключ Б*	никогда	никогда	блок для чтения-записи
1	0	1	ключ Б*	никогда	никогда	никогда	блок для чтения-записи
1	1	1	никогда	никогда	никогда	никогда	блок для чтения-записи

* если ключ Б может быть прочитан (из соответствующего прицепа), он не может служить для авторизации.

При попытке авторизоваться ключом Б, карта будет отвечать отказом в любом последующем доступе.

Mifare Plus

Смарт-карты Mifare Plus – улучшенная версия Mifare Classic, призваны повысить существующий уровень безопасности при использовании бесконтактных смарт-карт в различных

прикладных системах (оплата на транспорте, системы доступа, электронный сбор платы, парковки, интернет-кафе, программы лояльности и т.п.).

Карты Mifare Plus могут легко интегрироваться в существующие системы, где уже используются карты Mifare Classic. Уровень защищенности карт Mifare Plus может быть повышен в любой момент по мере развития системы путем активизации алгоритма AES-128 (Advanced Encryption Standard), обеспечивающего высокий уровень безопасности, целостности данных, аутентификации и шифрования.

Mifare Plus основана на глобальных открытых стандартах как по интерфейсу, так и по криптографии.

Mifare Plus имеет две версии:

- Mifare Plus S – это "Slim" версия, для быстрого перехода от Mifare Classic (может переключаться в режим SL1, SL3);
- Mifare Plus X – это "eXpert" версия, обеспечивающая большую гибкость, защищенность и функциональность (может переключаться в режим SL1, SL2, SL3).

Основные возможности

- 2 или 4 килобайта памяти (EEPROM);
- Простая структура фиксированной памяти, совместимая с Mifare Classic 1K и Mifare Classic 4K;
- Структура памяти идентична Mifare Classic 4K (сектора, блоки);
- Условия доступа свободно конфигурируемые;
- Поддержка ISO/IEC 14443-31 UID (4-байтовый UID, 4-байтовый NUID, 7-байтовый UID), опциональная поддержка случайных идентификаторов;
- Многосекторная аутентификация, многоблочное чтение и запись;
- AES-128 используется для обеспечения аутентификации и целостности;
- Анти-разрушающий механизм для записи ключей AES;
- Ключи могут храниться как ключи MIFARE CRYPTO1 (2 x 48 бит на сектор), так и как AES ключи (2 x 128 бит на сектор);
- Полная поддержка концепции виртуальной карты;
- Проверка близости (Proximity check);
- Скорость связи до 848 кбит/с;
- Количество операций одиночной записи: 200000 циклов (типичный);
- Сертификация по общим критериям: EAL4+ (Common Criteria Certification: EAL4+).

Уровни безопасности (Security level)

Карты, работающие на одном уровне безопасности, могут быть в любой момент переведены на более высокий уровень безопасности.

В прикладной системе карты Mifare Plus могут находиться только на одном конкретном уровне безопасности: SL1, SL2 или SL3. С завода-изготовителя чипы Mifare Plus (в картах, метках, браслетах и т.п.) поступают на уровне безопасности SL0.

Использовать в прикладной системе карты на уровне SL0 нельзя, чип Mifare Plus должен быть проинициализирован, т.е. переведен на уровень SL1, SL2 или SL3.

В прикладной системе можно последовательно повышать уровень безопасности, т.е. из SL1 переводить в SL2, а из SL2 переводит в SL3. Обратный перевод, с более высокого на более низкий уровень защиты, невозможен.

Уровень безопасности SL0

Начальный уровень. В этом состоянии чип Mifare Plus не хранит в себе никаких ключей. Из состояния SL-0 чип MIFARE Plus должен быть переведен на любой из трех более высоких уровней SL-1, SL-2 или SL-3. На уровне SL-0 чипы предварительно персонализируются по ключам, соответствующим Mifare Classic с использованием крипто-алгоритма CRYPTO1, и по ключам AES для работы с памятью.

Уровень безопасности SL1

На этом уровне карты Mifare Plus имеют полную совместимость с Mifare Classic.

Уровень безопасности SL2

Аутентификация по AES является обязательной. Для защиты данных используется CRYPTO1.

Уровень безопасности SL3

Для аутентификации, обмена и шифрования данных, для работы с памятью, а также для выявления удаленных атак по радиоканалу используется крипто-алгоритм AES.

Работа с памятью Mifare Plus

В основном структура памяти Mifare Plus SL3 идентична Mifare Classic (сектора, блоки), отличается структура прицепов (trailer), способ установки ключа авторизации и других ключей. Подробности выдаются NXP Semiconductors под NDA.

Mifare Ultralight

Phillips разработал карты Mifare Ultralight для использования со считывателями стандарта ISO/IEC14443A. Радиointерфейс (MIFARE RF) соответствует частям 2 и 3 стандарта ISO/IEC14443A. В основном Mifare Ultralight разработан для применения в сфере транспортных услуг в качестве бесконтактных билетов на одну поездку.

Основные возможности Mifare Ultralight

1. Радиочастотный интерфейс Mifare (ISO/IEC 14443A)
 - Бесконтактная передача данных и питание по радиоканалу (не требует батарей)
 - Рабочее расстояние до 100мм (зависит от геометрических параметров антенны)
 - Рабочая частота: 13,56МГц
 - Быстрая передача данных: 106Кбит/сек
 - Высокая надежность передачи (16-битовая CRC, проверка на четность...)
 - Настоящая антиколлизия (поддержка нескольких карт в поле одновременно)
 - 7-байтовый уникальный серийный номер
 - Время типовой транзакции менее 35мс
 - Быстрая транзакция счетчика: менее 10мс
2. Память
 - Размер 512бит, организована в виде 16 страниц, по 4 байта каждая
 - Возможность программирования постраничной блокировки записи
 - 32-битовая область "одноразового" программирования (для нужд пользователя) – OTP (One Time Programmable)
 - 384-битовая (48 байт) область чтения-записи (12 страниц)
 - Срок хранения данных – 5 лет
 - 10 000 циклов записи
3. Защита
 - Уникальный 7-байтовый серийный номер каждой карты
 - 32битовая область OTP
 - Функция блокировки записи отдельных страниц

Уникальный 7-байтовый UID однократно программируется в каждую карту в процессе производства и не может быть изменен в последствии, что является эффективной защитой от клонирования. Он может быть использован для организации криптозащиты хранимых данных. 32битовая область OTP (одноразового программирования) обеспечивает возможность однократной записи (т.е. данные, записанные в нее, не могут быть изменены впоследствии). Поле программируемой функции запрета записи соответствующих страниц. Эта функция позволяет уникально запрограммировать карту для специализированного применения.

Организация памяти Mifare Ultralight

512 бит перепрограммируемой памяти организовано в виде 16 страниц по 4 байта каждая.

80 бит зарезервировано под данные изготовителя.

16 бит предназначено для механизма блокировки, делающий доступ только для чтения.

32 бита доступны как OTP область.

384 бита используются для программирования памяти для чтения/записи.

В стертом состоянии ячейки читаются, как логический ноль, в записанном – как 1.

Byte Number	0	1	2	3	Page
Serial Number	SN0	SN1	SN2	BCC0	0
Serial Number	SN3	SN4	SN5	SN6	1
Internal / Lock	BCC1	Internal	Lock0	Lock1	2
OTP	OTP0	OTP1	OTP2	OTP3	3
Data read/write	Data0	Data1	Data2	Data3	4
Data read/write	Data4	Data5	Data6	Data7	5
Data read/write	Data8	Data9	Data10	Data11	6
Data read/write	Data12	Data13	Data14	Data15	7
Data read/write	Data16	Data17	Data18	Data19	8
Data read/write	Data20	Data21	Data22	Data23	9
Data read/write	Data24	Data25	Data26	Data27	10
Data read/write	Data28	Data29	Data30	Data31	11
Data read/write	Data32	Data33	Data34	Data35	12
Data read/write	Data36	Data37	Data38	Data39	13
Data read/write	Data40	Data41	Data42	Data43	14
Data read/write	Data44	Data45	Data46	Data47	15

Жирная рамка указывает на пользовательскую память.

1. UID/Серийный номер (SN0, SN1, SN2, SN3, SN4, SN5, SN6)

Уникальный 7-байтовый серийный номер и два байта контрольной суммы запрограммированы в первые 9 байт памяти. Поэтому он занимает страницы 0,1 и первый байт страницы 2. Второй байт страницы 2 зарезервирован для внутренних данных. Для обеспечения безопасности и системных требований эти байты защищены от записи после того, как были запрограммированы изготовителем при производстве.

2. Блокирующие байты (Lock0, Lock1)

Биты байтов 2 и 3 страницы 2 являются средством запрета записи соответствующих страниц. Каждая страница X из диапазона 3..15, может быть индивидуально заблокирована для записи путем установки соответствующего бита Lx в 1. После блокировки, страница становится доступна только для чтения.

- Другие опции
 - Режим пароля
 - Max Block функция
 - Режим Ответ-на-запрос (AOP)
 - Обратный вывод данных
 - Режим прямого доступа
 - Последовательность Терминатор(ов)
 - Защита от записи (через Лок-бит в блоке)
 - Метод быстрой записи (5 кбит/с по сравнению с 2 кбит/с)
 - ОТР Функциональность
 - POR задержки до 67 мс

T5557 является бесконтактным R/W идентификатором IC (IDIC) для применения в 125 кГц частотном диапазоне. 330-битная EEPROM (10 блоков, 33 бит каждый), позволяет читать и писать блоки. Блок 0 зарезервирован для настройки режимов работы T5557 метки. Блок 7 может содержать пароль для предотвращения несанкционированной записи.

Организация памяти Temic (T5557)

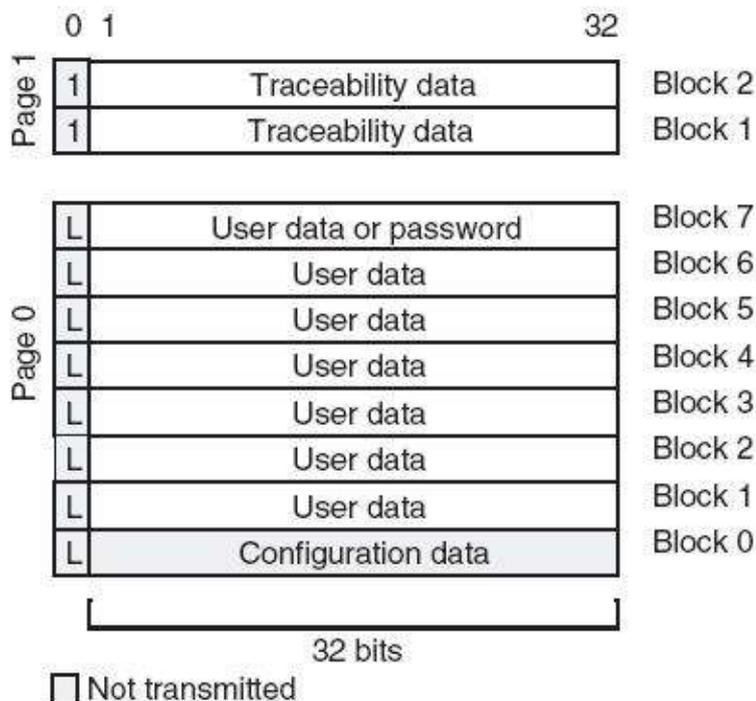
330 бит перепрограммируемой памяти организовано в виде 10 блоков по 4 байта каждый.

64 бит зарезервировано под данные производителя.

32 бит предназначено для настройки конфигурации T5557.

32 бита могут использоваться как пароль для предотвращения несанкционированной записи.

234(202 исключая пароль) бита используются для программирования памяти для чтения/записи.



1. Traceability data/Данные производителя (Page 1: block 1, block 2)

Данные страницы 1 содержат уникальный 5-байтовый серийный номер и защищены от записи производителем.

2. User data or password (Page 0: block 7)

Если T5557 в режиме пароля, то блок 7 содержит пароль для защиты от несанкционированной записи.

3. Блоки данных (Page 0: block 1 .. block6 (block 7))

Блоки 1..6 могут использоваться по усмотрению пользователя для чтения-записи. После производства страницы данных инициализированы "0".

4. Конфигурация (Page 0: block 0)

Блок 0 представляет собой регистр конфигурации T5557.

Конфигурация может быть настроена в режиме совместимости с e5550 или в расширенном режиме (X-Mode).

Конфигурация в режиме совместимости с e5550

0								1				2				3				← Байт													
7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	23	22	21	20	19	18	17	16	31	30	29	28	27	26	25	24	← Бит	
0	1	1	0	0	0	0	0	0	0	0					0								0							0	0		
Master Key <small>Note 1), 2)</small>				Data Bit Rate								Modulation				PSK- CF		AOR		MAX- BLOCK				PWD		ST-sequence Terminator		POR delay					
				RF/8 0 0 0 RF/16 0 0 1 RF/32 0 1 0 RF/40 0 1 1 RF/50 1 0 0 RF/64 1 0 1 RF/100 1 1 0 RF/128 1 1 1								0 0 0 0 0 Direct 0 0 0 0 1 PSK1 0 0 0 1 0 PSK2 0 0 0 1 1 PSK3 0 0 1 0 0 FSK1 0 0 1 0 1 FSK2 0 0 1 1 0 FSK1a 0 0 1 1 1 FSK2a 0 1 0 0 0 Manchester 1 0 0 0 0 Biphase (50) 1 1 0 0 0 Reserved				0 0 RF/2 0 1 RF/4 1 0 RF/8 1 1 Res.																	

Конфигурация в расширенном режиме (X-Mode)

0								1				2				3				← Байт													
7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	23	22	21	20	19	18	17	16	31	30	29	28	27	26	25	24	← Бит	
1	0	0	1	0	0	0	0								1																		
Master Key <small>Note 1), 2)</small>				Data Bit Rate RF/(2n+2)								Modulation				PSK- CF		AOR OTP		MAX- BLOCK				PWD		SST-Sequence Start Marker		Fast write		Inverse Data		POR-Delay	
				Direct PSK1 PSK2 PSK3 FSK1 FSK2 Manchester Biphase (50) Biphase (57)								0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 1 0 0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 0 1 1 0 0 0				0 0 RF/2 0 1 RF/4 1 0 RF/8 1 1 Res.																	

Считыватели

Модель	EM	HID	MUL	MC	MP	Temic	IL100	CAME
Настольные считыватели								
Z-2 RD_ALL	+	+	++	+	+	++		
Z-2 MF			++	++				
Z-2 MF-I			++	++	++			
Z-2 MF CCID			++	++	++			
Z-2 EHR	+	+				++	+	
RF-1996	+					++		
Z-2 Base								
Сетевые считыватели								
Matrix III RD-ALL	+	+	++	+				
Matrix III Net			++	++				
CP-Z-2 MF-I								
Matrix V	+						+	+

Обозначения:

+ - считыватель поддерживает чтение UID карты;

++ - считыватель поддерживает чтение и запись памяти карты.

Выделенные ячейки показывают поддерживает ли компонента то, что поддерживает считыватель.

Em – карта Em-Marine;

HID – карта HID;

MUL – карта Mifare Ultralight;

MC – карта Mifare Classic;

MP – карта Mifare Plus;

Temic – карта Temic (T5557, T5577);

IL100 – радио брелок IL-100;

CAME – радио брелок CAME.

Настольный считыватель Z-2 (мод. MF)/Z-2 USB MF



Основные возможности:

- Чтение идентификаторов: [Mifare Ultralight](#), [Mifare Classic](#)
- Чтение/запись памяти карт: [Mifare Ultralight](#), [Mifare Classic](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц
Чтение карт & брелков стандарта	Mifare Ultralight (чтение и запись), Mifare Classic (чтение и запись)
Дальность чтения	4 см

Питание	USB
Звуковая/световая индикация	сигнал зуммера, двухцветный светодиод
Рабочая температура	0°C до +50°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный
Выходной интерфейс	USB
Размер (мм)	110x80x25

Настольный считыватель Z-2 (мод. MF-I)



Основные возможности:

- Чтение идентификаторов: [Mifare Ultralight](#), [Mifare Classic](#), [Mifare Plus](#)
- Чтение/запись памяти карт: [Mifare Ultralight](#), [Mifare Classic](#), [Mifare Plus](#)
- Режим "Клавиатура"

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц
Чтение карт & брелков стандарта	Mifare Ultralight (чтение и запись), Mifare Classic (чтение и запись), Mifare Plus (чтение и запись)
Дальность чтения	2-6 см
Питание	USB
Звуковая/световая индикация	есть
Рабочая температура	+5°C до +50°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный
Выходной интерфейс	USB
Размер (мм)	110x80x25

Настольный считыватель Z-2 (мод. MF CCID)



Основные возможности:

- Чтение идентификаторов: Mifare ID, [Mifare Ultralight](#), [Mifare Classic](#), [Mifare Plus](#), Mifare DESfire, FeliCa/Felica lite
- Чтение/запись памяти карт: [Mifare Ultralight](#), [Mifare Classic](#), [Mifare Plus](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц
-----------------	-----------

Чтение карт & брелков стандарта	Mifare Ultralight (чтение и запись), Mifare Classic (чтение и запись), Mifare Plus (чтение и запись), Mifare DESfire, FeliCa/Felica lite
Поддерживаемые стандарты и операционные системы	Mifare Classic security (Crypto 1), PC/SC part 3 release 2.01.07, USB 2.0, поддержка CCID protocol, WIN XP,7,8, Linux. MAC OS
Дальность чтения	4-8 см
Питание	USB
Звуковая/световая индикация	есть
Рабочая температура	+5°C +40°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный
Выходной интерфейс	USB
Размер (мм)	110x80x25

Сетевой считыватель Matrix-III (мод. MF K Net) / Matrix-III Net



Основные возможности:

- Чтение идентификаторов: [Mifare Ultralight](#), [Mifare Classic](#)
- Чтение / запись памяти карт Mifare: [Mifare Ultralight](#), [Mifare Classic](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц
Чтение карт & брелков стандарта	Mifare Ultralight (чтение и запись), Mifare Classic (чтение и запись)
Дальность чтения	6 см
Звуковая/световая индикация	сигнал зуммера, двухцветный светодиод
Силовой выход	МДП-транзистор до 5А*
Ток коммутации	до 5А
Рабочая температура	-40°C +50°C
Материал корпуса	ABS пластик
Цвет корпуса	темно-серый металлик, светлый перламутр
Выходной интерфейс	Wiegand 26, Dallas Touch Memory (эмуляция DS1990A), RS-485
Размер (мм)	115x75x22

* совмещенные входы и выходы. Направление выхода выбирается на этапе конфигурирования устройства.

Сетевой считыватель CP-Z-2 (мод. MF-I)

**Основные возможности:**

- Чтение идентификаторов: [Mifare Ultralight](#), [Mifare Classic](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц
Чтение карт & брелков стандарта	Mifare Ultralight , Mifare Classic , Mifare ID
Дальность чтения:	1-4 см
Световая индикация:	есть
Подсветка:	Постоянно горит красный свет* (опционально)
Напряжение питания:	8 - 18 В постоянного тока
Потребление тока:	35mA
Рабочая температура:	-40°C +50°C
Материал корпуса:	Поликарбонат
Цвет корпуса:	Светло-серый, чёрный
Выходной интерфейс:	Dallas Touch Memory, Wiegand 26(34)
Максимальная длина линии от считывателя до контроллера:	Dallas Touch Memory: не более 15 м; Wiegand: не более 100 м
Размер (мм):	D45xH25

Настольный считыватель Z-2 (мод. RD_ALL)/Z-2 USB**Основные возможности:**

- Чтение идентификаторов: Em-Marine, HID ProxCard II, [Mifare UltraLight](#), [Mifare Classic](#), [Mifare Plus](#), [Temic](#), Cotag (опционально)
- Чтение/запись памяти карт: [Mifare Ultralight](#), [Temic](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц и 125 кГц одновременно
Чтение карт & брелков стандарта	EM Marine, HID ProxCard II, Mifare Classic , Mifare Plus , Mifare Ultralight (чтение и запись), Temic (чтение и запись), Cotag (опционально)
Дальность чтения	4-8 см
Питание	USB

Звуковая/световая индикация	есть
Рабочая температура	+5°C +40°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный
Выходной интерфейс	USB
Размер (мм)	110x80x25

Сетевой считыватель Matrix-III (мод. RD_All)



Основные возможности:

- Чтение идентификаторов: Em-Marine, HID ProxCard II, [Mifare UltraLight](#), [Mifare Classic](#), Mifare DESFire
- Чтение/запись памяти карты [Mifare Ultralight](#)

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	13,56 МГц и 125 кГц одновременно
Чтение карт & брелков стандарта	EM Marine, HID ProxCard II, Mifare, Mifare Ultralight (чтение и запись)
Дальность чтения	4-10 см
Напряжение питания	8 - 18 В постоянного тока
Потребление тока	135 мА
Звуковая/световая индикация	сигнал зуммера, двухцветный светодиод
Рабочая температура	-40°C +50°C
Материал корпуса	ABS пластик
Выходной интерфейс	RS-232, RS-485, Wiegand 26, Dallas Touch Memory
Размер (мм)	115x75x22

[Сетевая прошивка Matrix III Rd-All](#)

Настольный считыватель Z-2 (мод. E HTZ RF)/Z-2 EHR



Основные возможности:

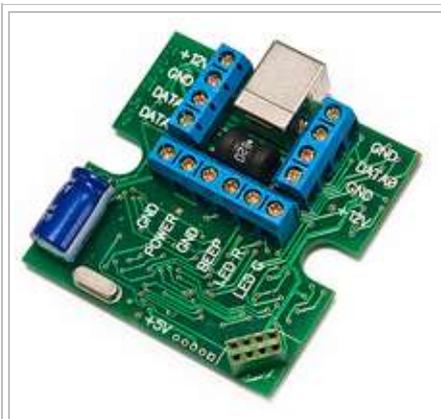
- Чтение номеров карт: iButton (DS1990, DS1996L), EM Marine, HID Prox II, [Temic](#)
- Чтение / запись памяти карты [Temic](#)
- Чтение радиобрелков 433Mhz (IL-100)
- Чтение/запись на ключ TM DS1996L

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	125 кГц и 433,92 МГц одновременно
Чтение карт & брелков стандарта	EM Marine, HID Prox II, Temic (чтение и запись) (дальность чтения/записи карт: 2-5 см)
Чтение радиобрелков	Keeloq 433 МГц - IL-100 (дальность чтения радиобрелков – до 10м)
Чтение ключей ТМ	DS1990 и DS1996L (чтение и запись)
Звуковая/световая индикация	есть
Рабочая температура	+5°C до +40°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный
Выходной интерфейс	USB
Размер (мм)	110x80x25

Настольный считыватель Z-1 (мод. N Z)/Z-2 Base



Основные возможности:

- Прямое программирование автономных контроллеров Z-5R/Z-5R (мод. Case) и Matrix-II (мод. E K) через контактное гнездо
- Программирование любых автономных контроллеров (других производителей), поддерживающих Dallas TM или Wiegand 26
- Получение и передача информации по линиям Dallas TM и Wiegand 26
- Управление командами из компьютера устройствами с логическими уровнями и питанием 12 вольтовых устройств
- Доступен выбор режима работы Comport или Клавиатура
- Доступно редактирование формата вывода номеров ключей

Страница на сайте IronLogic.ru

Спецификация

Напряжение питания	12 В
Потребление тока	не более 150 мА
Интерфейс связи с компьютером	USB (удалённость адаптера от компьютера не более 3 м)
Рабочая температура	+5°C +40°C
Размер (мм)	65x65x18

В режиме терминала получает от считывателя прочитанный номер по Wiegand 26 и Dallas TM (для считывателя представляется контроллером), передает контроллеру номер по Wiegand 26 и Dallas TM (для контроллера представляется считывателем, управляет внешней индикацией считывателей, управляет 12 В питанием (до 150 мА).

Сетевой считыватель Matrix-V (мод. E S RF) / Matrix-V

**Основные возможности:**

- "АнтиКлон" не позволяет дублировать радиобрелки (IL-100)
- Большая дальность чтения карточек EM-Marine
- абота с радиобрелком CAME серии Top

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	125 кГц, 433,92 МГц
Дальность чтения:	- карточка EM-Marine (05 ELR) – до 0,5 м - радиобрелок IL-100 (Keeloq) – до 10 м - радиобрелок CAME серии Top (TOP432NA и TOP434NA) – до 10 м
Напряжение питания:	12 В постоянного тока
Потребление тока:	500 мА
Звуковая/световая индикация	есть
Рабочая температура	-40°C +50°C
Выходной интерфейс	Dallas TM, Wiegand 26, RS-485
Цвет корпуса	темно-серый
Размер (мм)	230x230x35

Настольный считыватель Z-2 (мод. E NT Hotel)/Z-2 RF-1996**Основные возможности:**

- Чтение номеров карт: EM Marine, Temic
- Чтение/запись памяти карты Temic

Страница на сайте IronLogic.ru

Спецификация

Рабочая частота	125 кГц
Чтение карт & брелков стандарта	EM-Marine, Temic (чтение и запись) (дальность чтения/записи карт: 2-5 см)
Звуковая/световая индикация	есть
Рабочая температура	+5°C +40°C
Материал корпуса	ABS пластик
Цвет корпуса	чёрный

Выходной интерфейс	USB
Размер(мм)	110x80x25

Работа с сетевой прошивкой Matrix III Rd-All

Компонента ZR1C может передавать номера ключей от нескольких считывателей Matrix III Rd-All, подключенных к конвертеру ironlogic (Z-397, Z-397 Guard, Z-397 IP, Z-397 Web и т.д.).

Для этого нужно:

1. Если конвертер работает через USB (Z-397, Z-397 Guard), то подключить его к ПК и установить драйвер для конвертера (доступно на сайте [ironlogic](http://ironlogic.ru) в группе "Инструкции/Драйвера/Программы/Firmware"). Инструкция по установке в архиве с драйвером. После установки драйвера в Windows появится новый виртуальный com-порт (его номер можно посмотреть в Диспетчере устройств в разделе "Порты (Com и LTP)");
2. Если конвертер работает по IP (Z-397 Web и т.д.), то он должен быть в режиме "Сервер" (по умолчанию) (также может работать в режиме "Клиент");
3. Установить сетевую прошивку v409 или v410 (доступно на сайте [ironlogic](http://ironlogic.ru)) на каждый считыватель и определить серийный номер (с/н) каждого считывателя:
 1. Подключить один Matrix III Rd-All к конвертеру;
 2. Запустить программу прошивки от имени администратора и выбрать com-порт конвертера - начнется процесс прошивки;
 1. Если конвертер работает по IP, то нужно использовать программу Com2ip (доступно на сайте [ironlogic](http://ironlogic.ru)), которая создает виртуальный com-порт, к которому будет подключаться программа прошивки. В Com2ip нужно:
 1. Установить переключатель в значение "Клиент";
 2. Ввести IP-адрес конвертера;
 3. Ввести TCP порт, который задан в конфигурации конвертера (по умолчанию 1000, подробнее см. в инструкции конвертера);
 4. Установить флаг "Включен";
 3. Запомнить с/н считывателя, который можно посмотреть в программе прошивки (после завершения прошивки нажать кнопку "ASK", появится информация о считывателе "The MATRIX-III RD-ALL S/N:13566 [0501]...", где "13566" – с/н считывателя);
 1. Также с/н можно посмотреть с помощью ConfigZR1C.exe -> на странице "Тест" -> нажать кнопку "Загрузить" (маленькая, вторая снизу) -> нажать кнопку "Подключить..." -> в поле "Порт" ввести порт конвертера, например, "10.0.0.3:1000" или "COM3" -> нажать кнопку "Записать и закрыть" -> слева от кнопки "Подключить..." появится надпись "COM3 подключен – Matrix III Rd-All с/н 13566 v409", где "13566" – с/н считывателя;
 4. Прodelать то же самое со следующим считывателем;
 4. Подключить считыватели с сетевой прошивкой к конвертеру;
 5. Настроить формат передаваемых номеров ключей:
 1. В ZR1C с помощью метода InsertFormat(0, "Десятичный", "%.10u", "m01234567", 0, 0);
 2. В ConfigZR1C.exe на вкладке "Форматы":
 1. В таблице снять флаг со всех форматов или удалить их;
 2. Нажать кнопку "Добавить...", появится окно "Параметры формата";
 3. Установить флаг "Включен";
 4. Ввести в поле "Название:" любое название, например, "Десятичный";
 5. Ввести в поле "Шаблон:" и в поле "Параметры:" настройки формата (см. описание поля [Шаблон](#) и поля [Параметры](#)), например "%.10u" и "m01234567";
 6. В списке "Типы ключей" установить флаг "Неизвестный" (по умолчанию установлен);

7. Нажать кнопку "ОК";
8. На вкладке "Основное" установить флаг "Загрузить конфигурацию при старте ZR1C";
9. Нажать кнопку "Сохранить";
6. В компоненте ZR1C подключиться ко всем считывателям одним из способов:
 1. С помощью окна "Форма настройки драйвера" в ПО "1С Предприятие";
 2. С помощью метода "Open" / "Подключить":
 1. С помощью метода SetParameter("Port", "COM3\13566") установить имя порта в формате "COM3\13566", где "COM3" – com-порт конвертера, "13566" – с/н считывателя Matrix III Rd-All;
 1. Если конвертер работает по IP, то формат имени порта такой: "10.0.0.3:1000\13566", где "10.0.0.3" – IP-адрес конвертера, "1000" – TCP-порт, заданный в конфигурации конвертера, "13566" – с/н считывателя Matrix III Rd-All;
 2. Вызвать метод "Open" / "Подключить";
 3. Прodelать то же самое со следующим считывателями;
7. В "1С Предприятие" получать номера ключей, подносимых к считывателю, в событии ВнешнееСобытие (ExternalEvent), в котором:
 1. В поле "Источник" будет указано ID устройства (=имени порта, например, "COM3\13566"),
 2. В поле "Событие" будет значение "ДанныеКарты" (TracksData) (зависит от языка 1С Предприятие);
 3. В поле "Данные" будет номер ключа.

Вопросы и ответы

1. [Почему возникает ошибка "Формат не задан" при поднесении карты?](#)
2. [Как рассчитывается контрольная сумма Dallas?](#)
3. [Необходим шаблон "%u%u" \(параметры "b2 w01"\) с фиксированным числом разрядов, т.е. результат должен дополняться ведущими нулями до 20 разрядов](#)

1. Почему возникает ошибка "Формат не задан" при поднесении карты?

Проверьте выбран ли формат для всех фильтров ключей. В ConfigZR1C на вкладке "[Форматы](#)" на каждой вкладке фильтра выберите один формат с помощью переключателя во втором столбце "Выбор", на вкладке "[Основное](#)" установите флаг "Загрузить при старте ZR1C", нажмите кнопку "Сохранить", после этого перезагрузите компоненту, т.е. перезапустите 1С Предприятие.

2. Как рассчитывается контрольная сумма Dallas?

C++

```

BYTE GetDallasCS(const byte *Num, int Count)
{
    byte ret = 0;
    int i, j;
    const byte *p = Num;
    byte b;
    for (i = 0; i < Count; i++)
    {
        b = *p++;
        for (j = 0; j < 8; j++)
        {
            if (((ret ^ b) & 1) == 1)
                ret = (((ret ^ 0x18) >> 1) | 0x80);
            else
                ret >>= 1;
            b >>= 1;
        }
    }
}

```

```

    }
  }
  return ret;
}

```

Delphi

```

function GetDallasCS(Const Num: TBytes): Byte;
var
  i, j: Integer;
  b: Byte;
begin
  Result := 0;
  for i := 0 to Length(Num) - 1 do
  begin
    b := Num[i];
    for j := 0 to 7 do
    begin
      if ((b xor Result) and $1) <> 0 then
        Result := ((Result xor $18) shr 1) or $80
      else
        Result := Result shr 1;
      b := (b shr 1);
    end;
  end;
end;

```

где *Num* – массив байт номера ключа, *Count* – количество байт (обычно *Count*=7, значение первого (младшего) байта =0x01).

3. *Необходим шаблон "%u%u" (параметры "b2 w01") с фиксированным числом разрядов, т.е. результат должен дополняться ведущими нулями до 20 разрядов*

Например, параметры b2=230, w01=301, требуется результат 000000000000000230301.

- Шаблон: %.*u%u
- Параметры: 19-log₁₀(w01) b2 w01
- где
 - символ '*' в шаблоне означает, что количество разрядов будет указано в параметрах (перед параметром числа b2),
 - первый параметр "19-log₁₀(w01)" задаёт количество разрядов в первом числе (230), log₁₀(w01) = количество знаков во втором числе (301) - 1 = 2.

Пример для 1С

Код 1С:

```

ПодключитьВнешнююКомпоненту("с:\ZR1C.dll", "Comp", ТипВнешнейКомпоненты.Native);
Перем ДемоКомп;
ДемоКомп = Новый("AddIn.Comp.ZR1CExtension");

//ДемоКомп.ОчиститьЛог();
//ДемоКомп.ЗагрузитьНастройки();
//ДемоКомп.ОчиститьФорматы();

// Создаём новый формат

```

```
Перем FmtId;  
ДемоКомп.ВставитьФормат(0, "GuardSaaS", "%3u,%5u", "b2 w01", "", FmtId);  
// Задаём авто формат по умолчанию для всех считывателей  
//ДемоКомп.ИДФорматаПоУмолчанию = FmtId;  
  
// Задаём имя порта считывателя  
ДемоКомп.УстановитьПараметр("Порт", "COM4");  
// Задаём формат ключа для считывателя (не авто)  
ДемоКомп.УстановитьПараметр("ИДФорматаСчитывателя", FmtId);  
// Подключаемся к считывателю, в DevId получаем имя порта  
Перем DevId;  
ДемоКомп.Подключить(DevId);
```

где

"с:\ZR1C.dll" – путь к файлу компоненты;

"COM4" – com-порт для подключения к считывателю;

"%3u,%5u" и "b2 w01" – [шаблон](#) и [параметры](#) (как в PlaceCard) для форматирования и передачи номера от считывателя в 1С, этот формат автоматически добавляется при загрузке компоненты (его не нужно добавлять в коде 1С).

Методы [УстановитьПараметр](#), [ОчиститьФорматы](#), [ВставитьФормат](#) настраивают порт считывателя и формат номера ключа, эти настройки можно сохранить с помощью метода [СохранитьНастройки](#), и затем загрузить методом [ЗагрузитьНастройки](#).

Форматированный номер передается через событие "ВнешнееСобытие" в 1С.

Настройки компоненты также можно изменять с помощью программы "ConfigZR1C.exe".

Портативный режим

Компонента ZR1C (dll)

Не поддерживает портативный режим.

Настройки хранит всегда в реестре "HKCU\Software\IronLogic\ZR1C".

Лог файл "ZR1C.log" пишет в папку "%AppData%\IronLogic\ZR1C" (если запись лога включена).

Программа ConfigZR1C

Программа загружается в портативном режиме если в её папке присутствует файл "ConfigZR1C.xml" или "portable_mode_enabled".

В портативном режиме:

- Хранит настройки в файле "ConfigZR1C.xml" в своей папке;

В обычном режиме:

- Хранит настройки в файле "ConfigZR1C.xml" в папке "%AppData%\IronLogic\ZR1C".

Контакт с автором

E-mail: marketing@ironlogic.ru

Internet: www.ironlogic.ru

© 2016-2024 IronLogic