

# **Считыватель PR-G07**

**Руководство программиста**

Версия 1.1    октябрь 2014 г.

## Оглавление

---

Оглавление.....	1
История документа.....	5
Введение.....	6
Что нового в этом документе.....	6
Назначение.....	6
Совместимость.....	6
Принцип работы устройства.....	6
Время памяти тага.....	7
Память транзакций.....	7
Конструктивное исполнение.....	7
Физический интерфейс.....	8
Общие положения.....	8
Параметры интерфейса.....	8
Адресация.....	8
Формат пакетов.....	9
Физический уровень.....	9
Уровень приложения.....	9
Описание команд.....	11
Коды ошибок.....	11
Команды — утилиты.....	11
Команда R245_CMD_GET_VERSION.....	11
Назначение.....	11
Формат запроса.....	11
Формат ответа.....	11
Комментарии.....	12
Команда R245_CMD_SET_TIME.....	12
Назначение.....	12
Формат запроса.....	12
Формат ответа.....	12
Комментарии.....	12
Команда R245_CMD_SET_DATE.....	13
Формат запроса.....	13
Формат ответа.....	13
Комментарии.....	13
Команда R245_CMD_GET_TIME.....	13
Назначение.....	13
Формат запроса.....	13
Формат ответа.....	14
Комментарии.....	14
Команда R245_CMD_GET_DATE.....	14

Назначение.....	14
Формат запроса.....	14
Формат ответа.....	14
Комментарии.....	14
Команда R245_CMD_SET_ADDRESS.....	15
Назначение.....	15
Формат запроса.....	15
Формат ответа.....	15
Комментарии.....	15
Команда R245_CMD_GET_SENSORS.....	15
Назначение.....	15
Формат запроса.....	15
Формат ответа.....	16
Комментарии.....	16
Команды для работы с тагами.....	16
Команда R245_CMD_GET_NUM_TAGS_1.....	16
Назначение.....	16
Формат запроса.....	16
Формат ответа.....	16
Комментарии.....	17
Команда R245_CMD_GET_FIRST_TAG_1.....	17
Назначение.....	17
Формат запроса.....	17
Формат ответа.....	17
Комментарии.....	17
Команда R245_CMD_GET_NEXT_TAG_1.....	17
Назначение.....	17
Формат запроса.....	17
Формат ответа.....	18
Комментарии.....	18
Команда R245_CMD_GET_NUM_TAGS_2.....	18
Назначение.....	18
Формат запроса.....	18
Формат ответа.....	18
Комментарии.....	18
Команда R245_CMD_GET_FIRST_TAG_2.....	18
Формат запроса.....	19
Формат ответа.....	19
Комментарии.....	19
Команда R245_CMD_GET_NEXT_TAG_2.....	19
Назначение.....	19
Формат запроса.....	19
Формат ответа.....	19

Комментарии.....	19
Настройки радиоканалов.....	20
Команда R245_CMD_SET_DAMP_1.....	20
Назначение.....	20
Формат запроса.....	20
Формат ответа.....	20
Комментарии.....	20
Команда R245_CMD_SET_DAMP_2.....	20
Назначение.....	20
Формат запроса.....	20
Формат ответа.....	21
Комментарии.....	21
Команда R245_CMD_GET_DAMP_1.....	21
Назначение.....	21
Формат запроса.....	21
Формат ответа.....	21
Команда R245_CMD_GET_DAMP_2.....	21
Назначение.....	21
Формат запроса.....	21
Формат ответа.....	22
Команда R245_CMD_SET_TAG_MTIMEN_1.....	22
Назначение.....	22
Формат запроса.....	22
Формат ответа.....	22
Комментарии.....	22
Команда R245_CMD_SET_TAG_MTIMEN_2.....	22
Назначение.....	22
Формат запроса.....	22
Формат ответа.....	23
Комментарии.....	23
Команда R245_CMD_GET_TAG_MTIMEN_1.....	23
Назначение.....	23
Формат запроса.....	23
Формат ответа.....	23
Команда R245_CMD_GET_TAG_MTIMEN_2.....	23
Назначение.....	23
Формат запроса.....	23
Формат ответа.....	24
Команда R245_CMD_SET_CHAN_1.....	24
Назначение.....	24
Формат запроса.....	24
Формат ответа.....	24
Команда R245_CMD_SET_CHAN_2.....	24

Назначение.....	24
Формат запроса.....	24
Формат ответа.....	24
Работа с буфером транзакций.....	25
Команда R245_CMD_AUDIT_ON_OFF.....	25
Назначение.....	25
Формат запроса.....	25
Формат ответа.....	25
Комментарии.....	25
Команда R245_CMD_CLEAR_TRANS.....	25
Назначение.....	25
Формат запроса.....	25
Формат ответа.....	26
Комментарии.....	26
Команда R245_CMD_GET_NUM_TRANS.....	26
Назначение.....	26
Формат запроса.....	26
Формат ответа.....	26
Комментарии.....	26
Команда R245_CMD_GET_TRANSACT.....	26
Назначение.....	26
Формат запроса.....	26
Формат ответа.....	27
Комментарии.....	27
Команды конфигурирования считывателя.....	29
Команда R245_CMD_CONFIG_ACCESS.....	29
Назначение.....	29
Формат запроса.....	29
Формат ответа.....	29
Комментарии.....	29
Команда R245_CMD_READ_CFG_ACCESS.....	30
Назначение.....	30
Формат запроса.....	30
Формат ответа.....	30
Комментарии.....	31
Приложение 1.....	32
Вычисление контрольной суммы пакета.....	32
Для заметок.....	33

## История документа

---

Версия	Дата	Изменения
1.0	03.04.2009	Первая редакция документа
1.1	29.11.2014	Приведено в соответствие с реальной программой: <ul style="list-style-type: none"><li>1. Формирование пакетов RS-495</li><li>2. Порядок байтов при выдаче тагов из стека</li><li>3. Порядок байтов выдачи количества транзакций</li></ul>

## Введение

---

Данный документ предоставляет информацию, необходимую для непосредственной работы со считывателем дальней идентификации PR-G07 по интерфейсу RS-485. При этом в качестве ведущего может выступать любое устройство, в том числе контроллеры пользователя.

Не допускается использование кодов команд иных, чем описанные в данном документе.

### Что нового в этом документе

Данный документ представляет собой первую редакцию. Если вы нашли в документе опечатки или неточности, пожалуйста, сообщите о них по электронной почте: [sleo@relvest.ru](mailto:sleo@relvest.ru).

### Назначение

С использованием функций, описанных в документе, пользователь получает возможность работать с функционалом считывателя, недоступным при его использовании в системах доступа с интерфейсами *wiegand* или *Parsec*.

Если вы используете считыватель совместно с ПК, работающим под операционной системой Windows, то целесообразнее использовать динамическую библиотеку *Rx245.dll*, которая упрощает для программиста доступ к функционалу считывателя.

Данный документ позволяет подключать считыватели к любому микроконтроллерному устройству, имеющему интерфейс RS-485.

### Совместимость

Данный документ полностью соответствует функционалу считывателей версии 4.1 или старше. В последующих версиях считывателей описанные в данном документе функции даже при замене на новые будут оставлены для совместимости с ранее написанными пользовательскими приложениями.

### Принцип работы устройства

Считыватель предназначен для слежения за активными метками (тагами), попадающими в зону считывания. Считыватель может одновременно отслеживать до 64-х тагов, находящихся в зоне чтения каждого радиоканала. Дальность чтения в зависимости от условий эксплуатации и применяемой антенны составляет от 10 до 50 метров.

Для обеспечения требуемых характеристик при использовании в системах доступа (например, на автомобильных проходных) считыватель выполнен двухканальным. В общем случае каналы считывателя могут работать независимо, хотя для работы в составе систем доступа имеет мощный настраиваемый механизм межканального взаимодействия, позволяющий наряду с опросом датчиков (по два на канал) реализовывать сложные механизмы прохода или проезда.

### Время памяти тага

Обмен информацией между считывателем и тагом производится примерно два раза в секунду, но при этом возможны пропуски обменов за счет эфирных помех, коллизий (наложений обменов двух и более тагов в одно и то же время) и других факторов.

Для того, чтобы сделать слежение за тагами устойчивым, считыватель в каждом канале имеет встроенную память на 64 тага (стек текущих тагов). В этой памяти фиксируется каждый вновь появившийся таг, и для него заводится таймер, по обнулению которого таг считается вышедшим из зоны чтения (потерянным), и, соответственно, удаляется из памяти. Если таг с зафиксированным в памяти номером обнаружен считывателем до обнуления таймера, то таймер вновь перевзводится на полное время. Это время далее именуется **временем памяти тага в радиоканале**.

Данное время определяет помехозащищенность системы (чем больше время, тем меньше возникает ложных пропаданий тага с последующим его появлением). В то же время, чрезмерно большое время памяти повышает инерционность (в части фиксации выхода тага из зоны чтения). Данный параметр является программируемым для адаптации к конкретным условиям применения системы. На практике можно рекомендовать устанавливать это время не менее 10...20 периодов обмена тага со считывателем, то есть при периоде 0,5 секунды это время желательно устанавливать в пределах от 5 до 10 секунд.

В системе команд считывателя имеется функция для установки времени памяти тага в радиоканале.

### Память транзакций

Для использования в режиме мониторинга, а также для реализации функции аудита (использование в системе доступа) считыватель имеет встроенную энергонезависимую память событий (транзакций). Буфер транзакций организован в виде кольцевого буфера. Все факты появления и исчезновения тагов из поля считывателя, срабатывание датчиков автоматики и так далее заносится в эту память вместе с текущими датой и временем. Если кольцевой буфер переполняется, то первыми затираются наиболее старые события.

Размер кольцевого буфера составляет 1024 события. При получении транзакций от считывателя первыми передаются наиболее старые транзакции. После передачи транзакции ведущему (по его запросу) транзакция из памяти считывателя удаляется.

Буфер транзакций является общим для обоих каналов считывателя.

Для манипулирования с буфером транзакций имеется специальный набор функций в системе команд считывателя.

## **Конструктивное исполнение**

Считыватель выполнен в металлическом герметичном корпусе для уличного применения. Для подключения внешних антенн к обоим каналам имеется два антенных кабеля, выведенных через гермовводы.

Для подключения к внешнему оборудованию имеется также два сигнальных кабеля: через один осуществляется подача питания и подключение к контроллеру системы управления доступом, а через второй — подключение к хосту (RS-485) и подключение датчиков автоматики ворот (при использовании в системах доступа).



## Физический интерфейс

---

### Общие положения

В данном разделе рассмотрен формат пакетов, которыми обменивается считыватель с ведущим (например, с ПК) по интерфейсу RS-485.

Считыватель всегда является ведомым, компьютер (ПК или другое внешнее по отношению к считывателю устройство) – ведущим.

Ведущий формирует запросы к считывателю, а считыватель обязан на каждый запрос отправить ответ. Ниже рассмотрены форматы пакетов запроса мастера и пакеты ответов ведомого.

### Параметры интерфейса

Для работы со считывателем по интерфейсу RS-485 необходимо установить следующие параметры обмена:

- Скорость 9600 бод
- Длина данных 8 бит
- Один стоповый бит
- Без контроля четности

### Адресация

На одну линию RS-485 можно подключить до 30 считывателей (ограничение связано с физической нагрузочной способностью микросхем драйверов линии). При этом каждый считыватель должен иметь уникальный адрес в пределах от 1 до 254.

Нулевой адрес зарезервирован, а адрес 255 (0xFF) является общим адресом (broadcast). При передаче команды по общему адресу все считыватели принимают такую команду, но ни один из них не отвечает на команду.

Общий адрес может использоваться, например, для одновременной синхронизации встроенных часов реального времени, а также для смены адреса считывателя на новый известный, если текущий адрес считывателя по каким-то причинам не известен.

***Во время операции по смене адреса считывателя с использованием общего адреса на шине RS-485 должен быть подключенным только один считыватель!***

Смена текущего адреса считывателя производится программно, при этом считыватель запоминает свой адрес во встроенной энергонезависимой памяти.

## Формат пакетов

### Физический уровень

Обмен между ведущим и считывателем ведется пакетами, при этом со стороны ведущего формируется пакет запроса, а считыватель на него формирует пакет ответа. На физическом уровне пакет имеет нижеследующую структуру:

Номер байта	Ведущий	Ведомый	Комментарий
0	0xFB	0xFC	Символ начала пакета
1	0x??	0x??	Номер пакета
2	0x??	0x??	Адрес ведомого
3	0x00	0x00	Опции протокола – всегда 0x00
4	0x??	0x??	Команда или результат
5	0x??	0x??	Длина данных
			Опциональные данные (дина = M)
6 + M	0x??	0x??	Контрольная сумма (младший)
7 + M	0x??	0x??	Контрольная сумма (старший)
8 + M	0xFD	0xFE	Символ конца пакета

- ☐ Как видно из структуры пакетов, каждый пакет начинается символом начала и завершается символом конца. Данные значения зарезервированы только для этих целей. Если внутри пакета между символами начала и конца встречается байт со значением большим или равным, чем 0xFA, то такой байт заменяется на два байта [0xFA, X], где X есть заменяемый байт минус 0xF0. Замена байт осуществляется для данных всего пакета, включая поле контрольной суммы.
- ☐ Каждый пакет содержит двухбайтовое поле с контрольной суммой пакета. Контрольная сумма вычисляется для полей, отмеченных желтым цветом, **после** замены спецсимволов на двухбайтовые коды. Код для вычисления контрольной суммы приведен в Приложении 1. Начальное значение двухбайтовой контрольной суммы равно 0x6363. Контрольная сумма также формируется с заменой байтов более 0xF9 на двухбайтовые.
- ☐ Все запросы ведущего нумеруются (поле номера пакета), причем после успешного обмена с ведущим мастер инкрементирует номер пакета. Ведомый при ответе всегда ставит тот номер пакета, на который он отвечает.

### Уровень приложения

С точки зрения уровня приложения при дальнейшем рассмотрении системы команд считывателя нас интересуют только поля команды – результата, длины данных и самих данных. Все остальные поля, а также процедуры замены байтов с кодами от 0xFA по 0xFF на двухбайтовые комбинации находятся в компетенции драйвера линии и дальше рассматриваться не будут.

Формат запроса мастера (без протокольной «обвески») приведен ниже:

CmdRes	DLen	[Data]			
0x??	0x??				

Поле CmdRes при запросе мастера содержит код команды считывателю.

Поле DLen содержит длину данных (если команда не содержит данных, значение поля равно нулю)

Поле Data является опциональным и содержит число байтов, равное значению поля DLen

Ответ считывателя содержит аналогичные поля с той разницей, что при ответе поле CmdRes содержит код результата выполнения операции. При успешном выполнении значение поля CmdRes равно нулю.

## Описание команд

### Коды ошибок

Считыватель отвечает на любую корректно принятую адресованную ему команду (за исключением случая, когда команда передана по общему адресу). Нулевой результат (поле CmdRes) соответствует успешному выполнению операции, результат, отличный от нуля характеризует ошибку, возникшую во время выполнения операции считывателем. В случае ошибки никакие данные не возвращаются и поле DLen всегда имеет нулевое значение. Возможные коды ошибок приведены в таблице 1 ниже.

Таблица 1

Ошибка	Код	Описание
R245_RES_OK	00	Операция выполнена успешно
R245_ERR_WRONG_PARAM	20	Неправильное число параметров
R245_ERR_INVALID_DATA	21	Недопустимое значение данных
R245_ERR_NO_TAG	80	Нет тага в поле считывателя
R245_ERR_UNKNOWN	200	Неизвестная ошибка
R245_ERR_NO_IMPLEMENT	201	Команда не поддерживается

### Команды — утилиты

Данная группа команд предназначена для получения информации о версии , установки даты и времени и других операций общего назначения.

#### Команда R245\_CMD\_GET\_VERSION

##### Назначение

Команда предназначена для получения информации о текущей версии считывателя.

##### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	40 (0x28)
DLen	0
Data	нет

##### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	N (по длине строки данных)
Data	Текстовая строка с данными о считывателе Пример строки: «Reader 2x45 Ver. 4.5»

### Комментарии

1. Команда используется как справочная, для определения текущих параметров конкретного считывателя.
2. На момент написания документа текущая версия считывателя 4.5.

### Команда R245\_CMD\_SET\_TIME

#### Назначение

Команда предназначена для установки времени встроенных часов реального времени (RTC) считывателя.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	201 (0xC9)
DLen	4
Data	Текущее время для установки в RTC считывателя

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK при правильном формате команды, R245_ERR_WRONG_PARAM при неверной длине данных
DLen	0x00
Data	нет

### Комментарии

1. Если время не установлено, то содержимое часов реального времени не определено, и при считывании времени может получаться некорректный результат.
2. Формат времени выглядит следующим образом:

Байт	Значение
0	Текущее время, часы
1	Текущее время, минуты
2	Текущее время, секунды
3	День недели

### Команда R245\_CMD\_SET\_DATE

Команда предназначена для установки даты встроенных часов реального времени (RTC) считывателя.

#### **Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	202 (0xCA)
DLen	4
Data	Текущая дата для установки в RTC считывателя

#### **Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK при правильном формате команды, R245_ERR_WRONG_PARAM при неверной длине данных
DLen	0x00
Data	нет

#### **Комментарии**

1. Если дата не установлена, то содержимое RTC не определено, и при считывании даты может получаться некорректный результат.
2. Формат даты выглядит следующим образом:

Байт	Значение
0	Год, старший байт
1	Год, младший байт
2	Текущий месяц
3	День месяца (число)

### Команда R245\_CMD\_GET\_TIME

#### **Назначение**

Команда предназначена для считывания значения текущего времени из RTC считывателя.

#### **Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	203 (0xCB)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4
Data	4 байта с временем из RTC считывателя

### Комментарии

1. Если время не установлено, то содержимое часов реального времени не определено, и при считывании времени может получаться некорректный результат.
2. Формат времени выглядит следующим образом:

Байт	Значение
0	Текущее время, часы
1	Текущее время, минуты
2	Текущее время, секунды
3	День недели

### Команда R245\_CMD\_GET\_DATE

#### Назначение

Команда предназначена для считывания текущей даты из RTC считывателя.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	204 (0xCC)
DLen	0
Data	нет

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4
Data	4 байта с датой из RTC считывателя

### Комментарии

1. Если дата не установлена, то содержимое RTC не определено, и при считывании даты может получаться некорректный результат.
2. Формат даты выглядит следующим образом:

Байт	Значение
0	Год, старший байт
1	Год, младший байт
2	Текущий месяц
3	День месяца (число)

### Команда R245\_CMD\_SET\_ADDRESS

#### Назначение

Команда предназначена для смены текущего адреса считывателя на шине RS-485.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	30 (0x1E)
DLen	2
Data	Новый адрес считывателя в прямой и инверсной форме.

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	0

#### Комментарии

1. В команде адрес передается в прямой (первый байт) и инверсной форме (второй байт). Если адреса в прямой и инверсной формах не совпадают или адрес находится вне диапазона 0...254, считыватель выдает сообщение об ошибке и его адрес не меняется.
2. Если по какой-то причине адрес считывателя оказался неизвестным, то можно назначить считывателю новый адрес с использованием данной команды, отправленной по общему адресу. При этом считыватель на шине RS-485 во время этой процедуры должен быть только один.
3. Заводская установка адреса считывателя — 1.

### Команда R245\_CMD\_GET\_SENSORS

#### Назначение

Команда предназначена для получения информации о текущем состоянии входов датчиков автоматики.



### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	182 (0xB6)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4
Data	байт 0 — состояние датчика ворот канала 1; байт 1 — состояние датчика автомобиля кнала 1; байт 2 — состояние датчика ворот канала 2; байт 3 — состояние датчика автомобиля кнала .

### Комментарии

1. Состояние датчиков передается с учетом инверсии датчиков, установленной командой конфигурирования считывателя.

## Команды для работы с тагами

Данная группа команд позволяет получить доступ к оперативному стеку тагов радиоканалов считывателя. С помощью команд данной группы можно получить информацию о количестве находящихся в стеке (а значит в поле чтения радиоканалов) тагов, а также прочесть их серийные номера.

Следует иметь в виду, что информация в стеках меняется постоянно, и после получения информации и числе тагов в стеке попытка получить номера именно такого количества тагов может привести к несоответствию, так как за время чтения первых номеров содержимое стека может уже измениться.

### Команда R245\_CMD\_GET\_NUM\_TAGS\_1

#### Назначение

Команда предназначена для получения информации о количестве тагов в стеке первого радиоканала.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	80 (0x50)
DLen	0
Data	нет

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	1

Data

Число тагов в стеке

**Комментарии**

1. Данная информация может изменяться достаточно быстро, если в поле считывателя перемещается одновременно несколько тагов.

**Команда R245\_CMD\_GET\_FIRST\_TAG\_1****Назначение**

Команда предназначена для получения кода первого тага из находящихся в текущий момент в стеке первого радиоканала.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	81 (0x51)
DLen	0
Data	нет

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4 или 0, если нет тагов
Data	байт 0 — младший байт серийного номера тага, .... байт 3 — старший байт серийного номера тага.

**Комментарии**

1. Команда позиционирует внутренний указатель на начало стека, и должна обязательно выполняться перед получение остальных тагов из данного радиоканала.
2. Если в стеке тагов на текущий момент нет, то возвращается код ошибки R245\_ERR\_NO\_TAG

**Команда R245\_CMD\_GET\_NEXT\_TAG\_1****Назначение**

Команда предназначена для получения серийного номера очередного тага из стека первого радиоканала.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	82 (0x52)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4 или 0, если нет тагов
Data	Data    байт 0 — младший байт серийного номера тага, .... байт 3 — старший байт серийного номера тага.

### Комментарии

1. Если в стеке тагов на текущий момент нет, то возвращается код ошибки R245\_ERR\_NO\_TAG
2. Перед тем, как получать содержимое стека последовательными вызовами данной функции, необходимо спозиционировать внутренний указатель стека на первый таг с помощью предыдущей команды.

## Команда R245\_CMD\_GET\_NUM\_TAGS\_2

### Назначение

Команда предназначена для получения информации о количестве тагов в стеке второго радиоканала.

### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	83 (0x53)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	1
Data	Число тагов в стеке

### Комментарии

1. Данная информация может изменяться достаточно быстро, если в поле считывателя перемещается одновременно несколько тагов.

## Команда R245\_CMD\_GET\_FIRST\_TAG\_2

Команда предназначена для получения кода первого тага из находящихся в текущий момент в стеке второго радиоканала.

### Формат запроса

Запрос ведущего выглядит следующим образом:


CmdRes	84 (0x54)
DLen	0
Data	нет


### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4 или 0, если нет тагов
Data	Data    байт 0 — младший байт серийного номера тага, .... байт 3 — старший байт серийного номера тага.

### Комментарии

 Команда позиционирует внутренний указатель на начало стека, и должна обязательно выполняться перед получение остальных тагов из данного радиоканала.

 Если в стеке тагов на текущий момент нет, то возвращается код ошибки R245\_ERR\_NO\_TAG

## Команда R245\_CMD\_GET\_NEXT\_TAG\_2

### Назначение

Команда предназначена для получения серийного номера очередного тага из стека второго радиоканала.

### Формат запроса

Запрос ведущего выглядит следующим образом:


CmdRes	85 (0x55)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	4 или 0, если нет тагов
Data	Data    байт 0 — младший байт серийного номера тага, .... байт 3 — старший байт серийного номера тага.

### Комментарии

 Если в стеке тагов на текущий момент нет, то возвращается код ошибки R245\_ERR\_NO\_TAG



Перед тем, как получать содержимое стека последовательными вызовами данной функции, необходимо спозиционировать внутренний указатель стека на первый таг с помощью предыдущей команды.

## Настройки радиоканалов

Данная группа команд позволяет индивидуально настроить чувствительность радиоканалов, включить или выключить из работы каждый радиоканал, а также установить время памяти тагов в каналах.

### Команда R245\_CMD\_SET\_DAMP\_1

#### Назначение

Команда предназначена для установки затухания первого радиоканала. При минимальном затухании (параметр равен нулю) чувствительность радиоканала, а соответственно и дальность считывания, максимальны.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	12 (0x0C)
DLen	1
Data	затухание от 0 до 31

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

#### Комментарии



Если значение параметра выходит за границы 0...31, то считыватель возвращает код ошибки.



Установленное значение затухания радиоканала сохраняется в энергонезависимой памяти и восстанавливается после включения питания считывателя.

### Команда R245\_CMD\_SET\_DAMP\_2

#### Назначение

Команда предназначена для установки затухания второго радиоканала. При минимальном затухании (параметр равен нулю) чувствительность радиоканала, а соответственно и дальность считывания, максимальны.

#### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	13 (0x0D)
--------	-----------

DLen	1
Data	затухание от 0 до 31

#### **Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	
Data	

#### **Комментарии**

1. Если значение параметра выходит за границы 0...31, то считыватель возвращает код ошибки.
2. Установленное значение затухания радиоканала сохраняется в энергонезависимой памяти и восстанавливается после включения питания считывателя.

#### **Команда R245\_CMD\_GET\_DAMP\_1**

##### **Назначение**

Команда предназначена для получения установленного на данный момент затухания первого радиоканала считывателя.

##### **Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	18 (0x12)
DLen	0
Data	нет

##### **Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	1
Data	текущее затухание от 0 до 31

#### **Команда R245\_CMD\_GET\_DAMP\_2**

##### **Назначение**

Команда предназначена для получения установленного на данный момент затухания второго радиоканала считывателя.

##### **Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	19 (0x13)
--------	-----------

DLen	0
Data	нет

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	1
Data	текущее затухание от 0 до 31

**Команда R245\_CMD\_SET\_TAG\_MTIMEN\_1****Назначение**

Команда предназначена для установки времени памяти тагов в первом радиоканале считывателя.

**Формат запроса**

Запрос ведущего выглядит следующим образом:


CmdRes	25 (0x19)
DLen	2
Data	время памяти тага, младший байт первый.

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

**Комментарии**

 Установленное значение времени памяти тага сохраняется в энергонезависимой памяти и восстанавливается после включения питания считывателя.

**Команда R245\_CMD\_SET\_TAG\_MTIMEN\_2****Назначение**

Команда предназначена для установки времени памяти тагов во втором радиоканале считывателя.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	26 (0x1A)
DLen	2
Data	время памяти тага, младший байт первый.

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

**Комментарии**

1. Установленное значение времени памяти тага сохраняется в энергонезависимой памяти и восстанавливается после включения питания считывателя.

**Команда R245\_CMD\_GET\_TAG\_MTIMEN\_1****Назначение**

Команда позволяет получить установленное ранее время памяти тагов для стека первого радиоканала.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	27 (0x1B)
DLen	0
Data	нет

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	2
Data	время памяти тага, младший байт первый.

**Команда R245\_CMD\_GET\_TAG\_MTIMEN\_2****Назначение**

Команда позволяет получить установленное ранее время памяти тагов для стека второго радиоканала.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	28 (0x1C)
DLen	0
Data	нет



**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	2
Data	время памяти тага, младший байт первый.

**Команда R245\_CMD\_SET\_CHAN\_1****Назначение**

Команда предназначена для включения или отключения первого радиоканала считывателя.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	205 (0xCD)
DLen	1
Data	0 — выключить канал, 1 — включить.

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

**Команда R245\_CMD\_SET\_CHAN\_2****Назначение**

Команда предназначена для включения или отключения второго радиоканала считывателя.

**Формат запроса**

Запрос ведущего выглядит следующим образом:

CmdRes	206 (0xCE)
DLen	1
Data	0 — выключить канал, 1 — включить.

**Формат ответа**

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

## Работа с буфером транзакций

Считыватель может собирать все события (появление или пропадания из поля чтения тагов, срабатывание датчиков автоматики, принимаемые считывателем решения по отправке кода тага контроллеру) в специальный кольцевой буфер для последующего анализа. Для этого необходимо включить режим аудита. Ниже описаны команды, используемые при работе с буфером транзакций.

### Команда R245\_CMD\_AUDIT\_ON\_OFF

#### Назначение

Команда предназначена для включения и выключения функции аудита. При включенном аудите все события в считывателе протоколируются в энергонезависимом кольцевом буфере, откуда могут в дальнейшем быть считаны соответствующими командами.

#### Формат запроса

Запрос ведущего выглядит следующим образом:







CmdRes	181 (0xB5)
DLen	1
Data	0 (выключить) или 1 (включить)

#### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

#### Комментарии

-   Если режим аудита не включен, то буфер событий не формируется, и получить список событий соответствующими командами будет невозможно.
-   На функцию аудита, в том числе на выдачу событий ведущему, расходуется время микропроцессора, при этом скорость чтения тагов при большом их количестве может несколько снижаться.
-   Размер буфера транзакций равен 1000 событий. При переполнении буфера наиболее старые события затираются и становятся недоступными.

### Команда R245\_CMD\_CLEAR\_TRANS

#### Назначение

Команда предназначена для очистки буфера транзакций (событий) считывателя.

### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	120 (0x78)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	0
Data	нет

### Комментарии



Функция возвращает код результата R245\_RES\_OK, очищает кольцевой буфер транзакций и устанавливает начальные значения внутренних указателей.

## Команда R245\_CMD\_GET\_NUM\_TRANS

### Назначение

Команда предназначена для получения количества транзакций (событий), накопленных в буфере считывателя.

### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	121 (0x79)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	2
Data	2 байта с количеством транзакций в буфере, старший байт первым.

### Комментарии

1. Функция используется для получения числа транзакций на момент запроса. При извлечении транзакций следует руководствоваться длиной возвращаемой транзакции как признаком опустошения внутреннего буфера.

## Команда R245\_CMD\_GET\_TRANSACT

### Назначение

Команда предназначена для получения очередной транзакции из внутреннего буфера считывателя.

### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	122 (0x7A)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя выглядит следующим образом:

CmdRes	R245_RES_OK
DLen	16, 0 если нет транзакций
Data	16 байт с данными транзакции

Данные в транзакции декодируются следующим образом:

Байт	Значение	Комментарий
0	Код (младший)	Код транзакции (см. ниже)
1	Код (старший)	
2	Номер канала	
3	Резерв	Не используется
4	Код тага (младший)	Четыре байта серийного номера тага
5	Код тага	
6	Код тага	
7	Код тага (старший)	
8	День	Дата транзакции
9	Месяц	
10	Год (младший)	
11	Год (старший)	
12	Секунды	Время и день недели транзакции
13	Минуты	
14	Часы	
15	День недели	

### Комментарии

1. С кодом канала, равным нулю, возвращаются транзакции, относящиеся к считывателю (например, транзакция включения питания).

Ниже приведены коды транзакций считывателя:

Мнемоника	Код транзакции	Комментарий
-----------	----------------	-------------

trnPowerOn	0x01	Питание считывателя включено
trnPowerOff	0x02	Питание считывателя выключено
trnNewTag	0x10	Обнаружен новый таг в поле чтения
trnLossTag	0x11	Таг считывателем потерян
trnShockOn	0x20	Сработал вибродатчик
trnShockOff	0x21	Вибродатчик восстановлен
trnTamperOn	0x22	Сработал тампер корпуса тага
trnTamperOff	0x23	Тампер корпуса тага восстановлен
trnBatteryOff	0x24	Разряжена батарейка в таге
<b>Мнемоника</b>	<b>Код транзакции</b>	<b>Комментарий</b>
trnBatteryOn	0x25	Батарейка тага восстановлена
trnCarSensOn	0x101	Сработал датчик присутствия автомобиля
trnCarSensOff	0x102	Датчик присутствия автомобиля восстановлен
trnDcSensOn	0x103	Сработал датчик ворот (шлагбаума)
trnDcSensOff	0x104	Датчик ворот (шлагбаума) восстановлен
trnTagInOpposit	0x121	Текущий таг имеется на обработке в другом канале
trnChanLocked	0x122	Радиоканал заблокирован
trnChanUnLocked	0x123	Радиоканал разблокирован
trnCardSend	0x124	Карта направлена к контроллеру доступа
trnWaitCarSens	0x125	Ждем срабатывания датчика присутствия автомобиля
trnCarFound	0x126	Автомобиль появился в зоне датчика
trnCarNotFound	0x127	Автомобиль не появился в зоне датчика
trnGateNotOpened	0x128	Время вышло, ворота не открылись
trnWaitGateClose	0x129	Ожидаем закрывания ворот
trnAccessFinish	0x12A	Цикл доступа закончен, исходное состояние
trnWaitAccess	0x12B	Ожидание решения от контроллера доступа
trnWaitDefGate	0x12C	Ожидание времени открытых ворот
trnGateClosed	0x12D	Ворота закрылись
trnGateNotClosed	0x12E	Ворота не закрылись в установленное время

**Примечание:** В зависимости от версии считывателя, его режима работы и конфигурации могут появляться не все перечисленные выше транзакции.

## Команды конфигурирования считывателя

Команды данной группы позволяют сконфигурировать считыватель для адаптации к конкретным условиям эксплуатации: настроить реакцию на датчики автоматики, установить связанную работу каналов и так далее.

### Команда R245\_CMD\_CONFIG\_ACCESS

#### Назначение

Данная команда предназначена для комплексного конфигурирования логики работы каналов обработки тагов в режиме контроля доступа для автомобильных проходных. Поскольку каждый из устанавливаемых параметров отдельно не конфигурируется, необходимо иметь на стороне хоста полную копию конфигурации считывателя, либо получать ее из считывателя соответствующей командой, модифицировать необходимый параметр, а затем отправлять скорректированную конфигурацию обратно в считыватель.

#### Формат запроса

Запрос ведущего выглядит следующим образом:


CmdRes	150 (0x96)
DLen	16
Data	Конфигурация считывателя

#### Формат ответа

Ответ считывателя при правильном формате команды выглядит следующим образом:


CmdRes	R245_RES_OK
DLen	0
Data	нет


#### Комментарии


 Формат данных конфигурации представлен в таблице ниже:

Байт	Значение	Комментарий
0	Signature_1	0xAA
1	Signature_2	0x55
2	WaitCarTime_1	Время ожидания автомобиля после прочтения тага
3	WaitAccessTime_1	Время ожидания команды допуска от контроллера
4	GateDefaultTime_1	Время на открывание ворот первого канала
5	GateOpenTime_1	Время открытых ворот первого канала
6	Reserved_1	зарезервирован
7	AccessOptions_1	Битовые опции канала 1
8	WaitCarTime_2	Время ожидания автомобиля для второго канала
9	WaitAccessTime_2	Время ожидания доступа для второго канала
10	GateDefaultTime_2	Время на открывание ворот второго канала

Байт	Значение	Комментарий
11	GateOpenTime_2	Время открытых ворот второго канала
12	Reserved_2	зарезервирован
13	AccessOptions_2	Битовые опции канала 2
14	AccessMode	Алгоритм работы подсистемы доступа
15	WiegandInterval	Интервал вывода кодов тагов на интерфейс wiegand

 Байты сигнатуры служат как контрольные байты, подтверждающие, что весь пакет является конфигурацией считывателя.

 Все времена задаются в секундах.

 Байты AccessOptions\_ являются битовыми полями, при этом значение битов в каждом из байтов следующее:

Бит	Мнемоника	Битовая маска	Комментарий
0	USE_CAR_SENSOR	0x0001	Опрашивать датчик присутствия авто
1	USE_GATE_SENSOR	0x0002	Опрашивать датчик открытия ворот
2	CHECK_OPPOSIT_CHANNEL	0x0004	Использовать противоположный канал
3	CHANNEL_IS_ACTIVE	0x0008	Канал активен
4	CAR_SENS_INVERSION	0x0010	Инверсия датчика автомобиля
5	GATE_SENS_INVERSION	0x0020	Инверсия датчика ворот
6	CLEAR_GATE_ON_DC_OFF	0x0040	По закрыванию ворот сбрасывать время
7	Резерв		Не используется

## Команда R245\_CMD\_READ\_CFG\_ACCESS

### Назначение

Данная команда предназначена для комплексного конфигурирования логики работы каналов обработки тагов в режиме контроля доступа для автомобильных проходных. Поскольку каждый из устанавливаемых параметров отдельно не конфигурируется, необходимо иметь на стороне хоста полную копию конфигурации считывателя, либо получать ее из считывателя соответствующей командой, модифицировать необходимый параметр, а затем отправлять скорректированную конфигурацию обратно в считыватель.

### Формат запроса

Запрос ведущего выглядит следующим образом:

CmdRes	151 (0x97)
DLen	0
Data	нет

### Формат ответа

Ответ считывателя при правильном формате команды выглядит следующим образом:



CmdRes	R245_RES_OK
DLen	16
Data	Конфигурация считывателя

**Комментарии**

1. Формат данных конфигурации аналогичен передаваемым с использованием предыдущей команды (R245\_CMD\_CONFIG\_ACCESS).

## Приложение 1

### Вычисление контрольной суммы пакета

```
const static u16 CRCTBL[256] = {
    0x0000, 0xc0c1, 0xc181, 0x0140, 0xc301, 0x03c0, 0x0280, 0xc241,
    0xc601, 0x06c0, 0x0780, 0xc741, 0x0500, 0xc5c1, 0xc481, 0x0440,
    0xcc01, 0x0cc0, 0x0d80, 0xcd41, 0x0f00, 0xcfc1, 0xce81, 0x0e40,
    0x0a00, 0xcac1, 0xcb81, 0x0b40, 0xc901, 0x09c0, 0x0880, 0xc841,
    0xd801, 0x18c0, 0x1980, 0xd941, 0x1b00, 0xdbc1, 0xda81, 0x1a40,
    0x1e00, 0xdec1, 0xdf81, 0x1f40, 0xdd01, 0x1dc0, 0x1c80, 0xdc41,
    0x1400, 0xd4c1, 0xd581, 0x1540, 0xd701, 0x17c0, 0x1680, 0xd641,
    0xd201, 0x12c0, 0x1380, 0xd341, 0x1100, 0xd1c1, 0xd081, 0x1040,
    0xf001, 0x30c0, 0x3180, 0xf141, 0x3300, 0xf3c1, 0xf281, 0x3240,
    0x3600, 0xf6c1, 0xf781, 0x3740, 0xf501, 0x35c0, 0x3480, 0xf441,
    0x3c00, 0xfcc1, 0xfd81, 0x3d40, 0xff01, 0x3fc0, 0x3e80, 0xfe41,
    0xfa01, 0x3ac0, 0x3b80, 0xfb41, 0x3900, 0xf9c1, 0xf881, 0x3840,
    0x2800, 0xe8c1, 0xe981, 0x2940, 0xeb01, 0x2bc0, 0x2a80, 0xea41,
    0xee01, 0x2ec0, 0x2f80, 0xef41, 0x2d00, 0xedc1, 0xec81, 0x2c40,
    0xe401, 0x24c0, 0x2580, 0xe541, 0x2700, 0xe7c1, 0xe681, 0x2640,
    0x2200, 0xe2c1, 0xe381, 0x2340, 0xe101, 0x21c0, 0x2080, 0xe041,
    0xa001, 0x60c0, 0x6180, 0xa141, 0x6300, 0xa3c1, 0xa281, 0x6240,
    0x6600, 0xa6c1, 0xa781, 0x6740, 0xa501, 0x65c0, 0x6480, 0xa441,
    0x6c00, 0xacc1, 0xad81, 0x6d40, 0xaf01, 0x6fc0, 0x6e80, 0xae41,
    0xaa01, 0x6ac0, 0x6b80, 0xab41, 0x6900, 0xa9c1, 0xa881, 0x6840,
    0x7800, 0xb8c1, 0xb981, 0x7940, 0xbb01, 0x7bc0, 0x7a80, 0xba41,
    0xbe01, 0x7ec0, 0x7f80, 0xbf41, 0x7d00, 0xbdc1, 0xbc81, 0x7c40,
    0xb401, 0x74c0, 0x7580, 0xb541, 0x7700, 0xb7c1, 0xb681, 0x7640,
    0x7200, 0xb2c1, 0xb381, 0x7340, 0xb101, 0x71c0, 0x7080, 0xb041,
    0x5000, 0x90c1, 0x9181, 0x5140, 0x9301, 0x93c0, 0x5280, 0x9241,
    0x9601, 0x56c0, 0x5780, 0x9741, 0x5500, 0x95c1, 0x9481, 0x5440,
    0x9c01, 0x5cc0, 0x5d80, 0x9d41, 0x5f00, 0x9fc1, 0x9e81, 0x5e40,
    0x5a00, 0x9ac1, 0x9b81, 0x5b40, 0x9901, 0x99c0, 0x5880, 0x9841,
    0x8801, 0x48c0, 0x4980, 0x8941, 0x4b00, 0x8bc1, 0x8a81, 0x4a40,
    0x4e00, 0x8ec1, 0x8f81, 0x4f40, 0x8d01, 0x4dc0, 0x4c80, 0x8c41,
    0x4400, 0x84c1, 0x8581, 0x4540, 0x8701, 0x47c0, 0x4680, 0x8641,
    0x8201, 0x42c0, 0x4380, 0x8341, 0x4100, 0x81c1, 0x8081, 0x4040
};

void UpdateCRC(u16 *CRC, u8 Byte)
{
    *CRC = (u16) ((*CRC >> 8) & 0xff) ^ CRCTBL[(*CRC ^ Byte) & 0xff];
}

u16 CalcCRC(u16 InitCRC, void * Data, u32 Len)
{
    u16 CRC;
    u8* u8Data;
    u8Data = Data;
    CRC = InitCRC;
    while (Len--) {
        UpdateCRC(&CRC, *u8Data++);
    }
    return CRC;
}

crc = CalcCRC(INITIAL_CRC, packet_buffer, packet_length );
```

## **Для заметок**

---