ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ГОСТ Р 58666— 2019 (ИСО/МЭК 18000-3:2010)**

## Информационные технологии

# ИДЕНТИФИКАЦИЯ РАДИОЧАСТОТНАЯ ДЛЯ УПРАВЛЕНИЯ ПРЕДМЕТАМИ

## Параметры радиоинтерфейса для связи на частоте 13,56 МГц

(ISO/IEC 18000-3:2010, MOD)

Издание официальное

Москва
Стандартинформ
2020

gost.e-ecolog.ru

## Предисловие

1 ПОДГОТОВЛЕН Акционерным обществом «Всероссийский научно-исследовательский институт сертификации» (АО «ВНИИС»), Некоммерческим партнерством «Русское общество содействия развитию биометрических технологий, систем и коммуникаций» (Некоммерческое партнерство «Русское биометрическое общество»), Федеральным государственным бюджетным учреждением «Государственная публичная научно-техническая библиотека России» (ГПНТБ России), Некоммерческим партнерством «Международный Центр Трансфера Технологий» (НП «МЦТТ»), Закрытым акционерным обществом «3М Россия» (ЗАО «3М Россия») и Обществом с ограниченной ответственностью «Спектрум Менеджмент» (ООО «Спектрум Менеджмент») на основе официального перевода на русский язык англоязычной версии указанного в пункте 4 стандарта, который выполнен Некоммерческим партнерством «Русское биометрическое общество», ГПНТБ России, НП «МЦТТ» и ЗАО «3М Россия»

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 355 «Технологии автоматической идентификации и сбора данных»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 19 ноября 2019 г. № 1185-ст

4 Настоящий стандарт является модифицированным по отношению к международному стандарту ИСО/МЭК 18000-3:2010 «Информационные технологии. Идентификация радиочастотная для управления предметами. Часть 3. Параметры радиоинтерфейса для связи на частоте 13,56 МГц» (ISO/IEC 18000-3:2010 «Information technology — Radio frequency identification for item management — Part 3: Parameters for air interface communications at 13,56 MHz», MOD), путем изменения отдельных фраз (слов, значений показателей, ссылок), которые выделены в тексте курсивом. Внесение указанных технических отклонений направлено на учет потребностей национальной экономики Российской Федерации.

Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте, приведены в дополнительном приложении ДА.

Сведения о соответствии международных и русских команд, параметров и технических терминов, использованных в настоящем стандарте, приведены в дополнительном приложении ДБ.

Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта приведено в дополнительном приложении ДВ

5 ВВЕДЕН ВПЕРВЫЕ

6 Некоторые элементы настоящего стандарта могут быть объектами патентных прав. Международная организация по стандартизации (ИСО) и Международная электротехническая комиссия (МЭК) не несут ответственности за установление подлинности каких-либо или всех таких патентных прав

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© ISO, 2010 — Все права сохраняются
© Стандартинформ, оформление, 2020

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

## Содержание

1 Область применения . . . . . 1
2 Требования соответствия . . . . . 1
3 Нормативные ссылки . . . . . 1
4 Термины и определения . . . . . 2
5 Обозначения и сокращения . . . . . 3
5.1 Обозначения . . . . . 3
5.2 Сокращения . . . . . 4
5.3 Примечания . . . . . 5
6 Требования: физический уровень, система управления коллизиями и протокол передачи данных для систем, работающих на частоте 13,56 МГц . . . . . 5
6.0 Общие положения, применимые ко всем МОДА . . . . . 5
6.1 МОДА 1: физический уровень, система управления коллизиями и протоколы МОДА 1 . . . . . 6
6.2 МОДА 2: физический уровень, система управления коллизиями и протоколы МОДА 2 . . . . . 13
6.3 МОДА 3: физический уровень, система управления коллизиями и протоколы МОДА 3 . . . . . 49
7 Маркировка оборудования . . . . . 121
Приложение А (справочное) МОДА 1: обязательные и дополнительные команды, требующиеся для поддержки протокола передачи данных по *[3]* . . . . . 122
Приложение В (справочное) МОДА 2 и МОДА 3: фазовая модуляция колебаний . . . . . 123
Приложение С (обязательное) МОДА 3: таблицы перехода состояний . . . . . 126
Приложение D (обязательное) МОДА 3: таблицы ответов на команды . . . . . 135
Приложение Е (обязательное) МОДА 3: коды ошибок . . . . . 143
Приложение F (обязательное) МОДА 3: счетчик слота . . . . . 144
Приложение G (справочное) МОДА 3: пример алгоритма выбора параметра *Q* счетчика слота . . . 145
Приложение Н (справочное) МОДА 3: пример инвентаризации и доступа к радиочастотной метке . 146
Приложение I (справочное) МОДА 3: расчет 5-битового и 16-битового циклического избыточного кода . . . . . 148
Приложение J (справочное) МОДА 3: метод ASK: Модуляция сигнала в канале связи «УСО-радиочастотная метка» . . . . . 150
Приложение К (справочное) МОДА 3: пример обмена потоками данных . . . . . 152
Приложение L (справочное) МОДА 3. функции радиочастотной метки . . . . . 155
Приложение М (справочное) Циклический избыточный код (CRC) (16 бит) . . . . . 156
Приложение N (справочное) Циклический избыточный код (CRC) для МОДА 2 (32 бита) . . . . . 158
Приложение О (справочное) Известные возможные перекрестные взаимодействия между МОДА, определенными в настоящем стандарте . . . . . 161
Приложение ДА (справочное) Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте . . . . . 162
Приложение ДБ (справочное) Сведения о соответствии международных и русских команд, параметров и технических терминов . . . . . 163
Приложение ДВ (справочное) Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта . . . . . 166
Библиография . . . . . 167

## Введение

Серия стандартов ГОСТ Р ИСО/МЭК 18000 «Информационные технологии. Идентификация радиочастотная для управления предметами» подготовлена с целью:

- определить общие протоколы для связи в полосах радиочастот, используемых на международном уровне для радиочастотной идентификации (РЧИ), и, по возможности, определить использование одних и тех же протоколов для всех полос радиочастот так, чтобы уменьшить проблемы перехода от одной полосы радиочастот к другой;
- свести к минимуму стоимость программного обеспечения и затраты на внедрение системы;
- добиться, чтобы управление, контроль и обмен информацией были общими настолько, насколько это возможно.

Она состоит из следующих частей:

- Часть 2. Параметры радиоинтерфейса для частот ниже 135 кГц;
- Часть 3. Параметры радиоинтерфейса для связи на частоте 13,56 МГц;
- Часть 4. Параметры радиоинтерфейса для связи на частоте 2,45 ГГц;
- Часть 6. Параметры радиоинтерфейса для связи в диапазоне частот от 860 до 960 МГц. Общие требования;
- Часть 7. Параметры активного радиоинтерфейса для связи на частоте 433 МГц;
- Часть 61. Параметры радиоинтерфейса для связи в диапазоне частот от 860 до 960 МГц, тип A;
- Часть 62. Параметры радиоинтерфейса для связи в диапазоне частот от 860 до 960 МГц, тип B;
- Часть 63. Параметры радиоинтерфейса для связи в диапазоне частот от 860 до 960 МГц, тип C;
- Часть 64. Параметры радиоинтерфейса для связи в диапазоне частот от 860 до 960 МГц, тип D.

Настоящий стандарт предусматривает 3 МОДА, ориентированных на различные области применения. Подробные технические различия между МОДА приведены в таблицах параметров.

Настоящий стандарт касается исключительно систем, работающих на частоте 13,56 МГц.

Международная организация по стандартизации (ИСО) и Международная электротехническая комиссия (МЭК) обращают внимание на тот факт, что следование этому стандарту может затронуть патентные права.

ИСО и МЭК не несут ответственности за установление подлинности каких-либо или всех патентных прав.

Патентообладатели заверили ИСО и МЭК, что они готовы вести переговоры с пользователями по всему миру о лицензировании на разумных и непредвзятых условиях. В этом отношении, заявления патентообладателей зарегистрированы в ИСО и МЭК. Информация может быть получена от следующих компаний:

| Контактная информация | Номер патента |
|---|---|
| EM Microelectronic SA<br>Mr Marc Degrauwe, IP manager<br>Rue des Sors 3<br>CH-2074 Marin, Switzerland<br>(T) +41 32 755 51 11<br>(F) +41 32 755 54 03<br>info@ emmicroelectronic.com<br>gost.e-ecolog.ru | EP 97 115772.2<br>EP 0 902 546<br>US 151803 |
| Impinj, Inc.<br>Chris Diorio, CTO<br>701 N 34th Street, suite 300<br>Seattle, WA 98103, USA<br>(T) +1 206 834 1115<br>(F) +1 206 517 5262<br>gost.e-ecolog.ru<br>gost.e-ecolog.ru | EP 97 115772.2<br>EP 0 902 546<br>US 151803 |

*Продолжение таблицы*

| Контактная информация | Номер патента |
|---|---|
| Intermec IP Corporation<br>Phyllis T. Turner-Brim, Esq.<br>Legal Department<br>6001 — 36th Avenue West<br>Everett, WA 98203, USA<br>(T) +1 425-265-2480<br>(F) +1 425-501-6587<br>phyllis.turnerbrim@intermec.com | US 5673037<br>DE 69530547.6<br>EP 0702323<br>FR EP0702323<br>GB EP0702323<br>KR 204748<br>TW 318306<br>US 6172596<br>US 6400274 (claims 1-10 only)<br>US 6404325<br>US 5550547<br>TW 307079<br>KR 210830<br>US 5521601<br>US 5777561<br>US 5828318<br>EP 1020044<br>US 5912632<br>US 5942987<br>TW 352492<br>KR 244844<br>US 5995019<br>US 6400274 (claims 11 et. Seq.)<br>US 6288629<br>US 6812841<br>US 6812852<br>US 7427912 |
| Magellan Technology Pty. Limited<br>IP Manager<br>65 Johnston St<br>Annandale, NSW 2038, Australia<br>(T) +61 2 9562 9800<br>(F) +61 2 9518 7620<br>info@magellan-technology.com | US5302954<br>SG37971<br>DE3854478D<br>EP0390822<br>US5485154<br>US10/927,957<br>US6967573<br>JP2002500465T<br>JP2006-180816<br>DE69835452<br>EP1048126<br>AU2006202886<br>AU785098<br>US7248145<br>US7259654<br>US11/538,271<br>US11/538/242<br>JP2003 526148<br>JP2006 344227<br>DE60119910<br>EP1266458<br>EP07013773<br>EP1544782<br>EP1544788<br>EP1679635 |

*Окончание таблицы*

| Контактная информация | Номер патента |
|---|---|
| NXP B.V.<br>Marc Schouten<br>Intellectual Property & Licensing<br>High Tech Campus 32<br>5656 AE Eindhoven<br>The Netherlands<br>(T) +31 40 27 26951<br>(F) +31 40 27 42640<br>marc.schouten@nxp.com | |
| TAGSYS, SA<br>Alastair McArthur, CTO<br>180, Chemin de Saint Lambert<br>13821 La Penne sur Huveaune<br>FRANCE<br>+33 491 27 57 00<br>+33 49t 27 57 01<br>alastair.mcarthur@<br>tagsysrfrd.com.au<br>gost.e-ecolog.ru | US 6,641,036<br>EP 1232471<br>US 6992567<br>EP 1256083<br>US 6946951<br>EP 1358644<br>US 6538564<br>EP 953181 |
| Texas Instruments Inc.<br>Robby Holland, Licensing Manager<br>P.O. Box 655464, MS 3999<br>Dallas TX 75256<br>(T) +1 972 917 4367<br>(F) +1 972 917 4418<br>r-holland3@ti.com | EP1 038257<br>US 09/315708<br>JP 00-560700<br>EP 1 034644<br>US 6442215<br>CN 1273730A<br>WO00/04686<br>EP 0669591B<br>AT-PS 401127 |
| Zebra Technologies Corporation<br>Eric McAlpine, IP Counsel<br>Legal Department<br>333 Corporate Woods Parkway<br>Vernon Hills, IL 60061-3109<br>(T) +1 847 793 5640<br>(F) +1 847 955 4514<br>emcalpine@zebra.com | US 6784787<br>EP 1031046<br>EP 1291671<br>EP 05017862.3<br>US 5680459<br>US 5557280<br>US 5699066<br>EP 0585132<br>US 6198381<br>JP 10-272945<br>US 5537105<br>US 5966083<br>US 5995017 |

Следует обратить внимание на то, что некоторые элементы этого стандарта могут быть объектом патентных прав, не указанных выше. ИСО и МЭК не несут ответственности за выявление каких-либо или всех патентных прав. Последнюю информацию по вопросам патентования в ИСО можно найти по адресу: gost.e-ecolog.ru .

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Информационные технологии**

# ИДЕНТИФИКАЦИЯ РАДИОЧАСТОТНАЯ ДЛЯ УПРАВЛЕНИЯ ПРЕДМЕТАМИ

**Параметры радиоинтерфейса для связи на частоте 13,56 МГц**

Information technology. Radio frequency identification for item management. Part 3. Parameters for air interface communications at 13,56 MHz

**Дата введения — 2020—06—01**

## 1 Область применения

Настоящий стандарт содержит рекомендации, касающиеся физического уровня, системы управления коллизиями и протоколов систем радиочастотной идентификации, предназначенных для идентификации предметов и работающих на частоте 13,56 МГц в соответствии с требованиями *[1]*.

Настоящий стандарт содержит описание систем для каждой МОДА*, определенной в разделе 6.

Настоящий стандарт определяет три, не создающих существенных взаимных помех, МОДА.

МОДА несовместимы.

Несовместимые МОДА не оказывают воздействие друг на друга.

## 2 Требования соответствия

Для того чтобы претендовать на соответствие настоящему стандарту, необходимо соблюдение всех его положений, за исключением положений с пометкой «опционально». Кроме того, необходимо действовать в соответствии с национальными требованиями использования радиочастотного спектра (что может потребовать дополнительных ограничений).

Соответствующие методы испытаний определены в *ГОСТ Р 56914*.

## 3 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р ИСО/МЭК 7816-6—2013 Карты идентификационные. Карты на интегральных схемах. Часть 6. Межотраслевые элементы данных для обмена

ГОСТ Р ИСО/МЭК 15693-1—2013 Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 1. Физические характеристики

ГОСТ Р ИСО/МЭК 15693-2—2013 Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 2. Воздушный интерфейс и инициализация

ГОСТ Р ИСО/МЭК 15693-3—2011 Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 3. Антиколлизия и протокол передачи данных

ГОСТ Р ИСО/МЭК 15963 Информационные технологии. Радиочастотная идентификация для управления предметами. Уникальная идентификация радиочастотных меток

ГОСТ Р ИСО/МЭК 18046 Автоматическая идентификация. Идентификация радиочастотная. Методы испытаний технических характеристик устройств радиочастотной идентификации

* *МОДА (Mode) радиочастотной идентификации — одна из различных систем радиочастотной идентификации, работающих в той же полосе частот, которые могут быть или не быть совместимыми, но не создают друг другу существенные помехи.*

**Издание официальное**

ГОСТ Р ИСО/МЭК 19762-1 Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 1. Общие термины в области АИСД

ГОСТ Р ИСО/МЭК 19762-2 Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 2. Оптические носители данных (ОНД)

ГОСТ Р ИСО/МЭК 19762-3 Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 3. Радиочастотная идентификация (РЧИ)

ГОСТ Р ИСО/МЭК 19762-4 Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 4. Общие термины в области радиосвязи

ГОСТ Р 56914 Информационные технологии. Методы испытаний на соответствие устройств радиочастотной идентификации. Часть 3. Методы испытаний радиоинтерфейса для связи на частоте 13,56 МГц

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

## 4 Термины и определения

В настоящем стандарте применены термины по *ГОСТ Р ИСО/МЭК 19762-1*, *ГОСТ Р ИСО/МЭК 19762-3*, *ГОСТ Р ИСО/МЭК 19762-4*, а также следующие термины с соответствующими определениями:

4.1 **закодированный текст** (cover-coded text): Информация, которая зашифрована методом защиты кодированием.

4.2 **защита кодированием** (cover-coding): Метод закрытия устройством считывания/опроса (УСО) информации, которая передается радиочастотной метке.

4.3 **дуплексная связь** (full-duplex communications): Передача данных (в обоих направлениях) при наличии активирующего поля, излучаемого трансивером.

4.4 **полудуплексная связь** (half-duplex communications): Поочередная передача данных в любом направлении, которая ведется в каждый момент времени только в одном направлении.

4.5 **маркер подлинности** (handle, descriptor): RN16, 16-битовое случайное число, которое служит для установления подлинности (опознания, аутентификации) радиочастотных меток в состояниях **open** и **secured**.

4.6 **код PacketCRC** (PacketCRC): 16-битовый циклический избыточный код (CRC), который радиочастотная метка с ненулевым значением *XI* (индикатора расширенного протокола управления, см. 5.1), динамически вычисляет на основе значений *PC*, *XPC* и *UII*, и передает путем модуляции нагрузкой во время инвентаризации* (см. 4.13).

4.7 **данные PacketPC** (*PacketPC*): Информация о протоколе управления, которую радиочастотная метка с ненулевым значением *XI* динамически вычисляет и передает путем модуляции нагрузкой во время инвентаризации (см. 4.14).

4.8 **фазовая модуляция колебаний** (phase jitter modulation (PJM)): Метод модуляции, при котором данные передаются в виде малых изменений фазы силового питающего поля.

4.9 **физический уровень** (physical layer): Кодирование данных путем модуляции формы радиоволн, используемые при обмене данными между устройством считывания/опроса (УСО) и радиочастотной меткой при передаче данных в обоих направлениях.

---

* *Инвентаризация (inventory) — операция идентификации радиочастотной метки устройством считывания/опроса (УСО).*

4.10 **опорный интервал** (pivot): Средняя длительность передаваемого символа данных при передаче в направлении «УСО — радиочастотная метка». R=>T: Опорный интервал = (длительность 0 + длительность 1)/2. См. Направление передачи от УСО к радиочастотной метке (R => T) в 5.1.

4.11 **открытый текст** (plaintext): Информация, которая не защищена кодированием.

4.12 **ввод в действие** (recommissioning): Значительные изменения функциональности радиочастотной метки и/или содержимого памяти, по командам УСО, обычно в результате изменения модели использования или назначения радиочастотной метки.

4.13 **код StoredCRC** (StoredCRC): 16-битовый циклический избыточный код (CRC), который вычисляется радиочастотной меткой для *данных StoredPC* и *UII*, сохраняется в памяти UII при включении питания, и может передаваться в ответе при инвентаризации (см. 4.13).

4.14 **данные StoredPC** (StoredPC): Данные, определяющие вид протокола управления (protocol-control information), хранящаяся в памяти UII, которую радиочастотная метка с нулевым значением *XI* передает путем модуляции нагрузкой во время инвентаризации (см. 4.7).

4.15 **интервал Tari** (Tari): Опорный интервал при передаче нулевого бита данных в канале связи «УСО-радиочастотная метка».

## 5 Обозначения и сокращения

В настоящем стандарте использованы как собственные символьные обозначения и сокращения, так и принятые в *ГОСТ Р 56914, ГОСТ Р ИСО/МЭК 19762-1, ГОСТ Р ИСО/МЭК 19762-3, ГОСТ Р ИСО/МЭК 19762-4*.

### 5.1 Обозначения

DR — результат деления (отношение) (для метода ASK) или 0 бит выбора канала для ответа (для метода PJM);

$F_c$ — несущая частота;

M(ASK) — тип модуляции ответа радиочастотной метки;

$M_h$ — огибающая радиочастотного (РЧ) сигнала пульсации (верхняя часть);

$M_l$ — огибающая радиочастотного (РЧ) сигнала пульсации (нижняя часть);

M(PJM) — бит 1 и бит 2 выбора канала для ответа;

$M_s$ — уровень радиочастотного РЧ сигнала в состоянии ВЫКЛЮЧЕНО;

Q — параметр количества слотов (параметр, который УСО использует для регулировки вероятности ответа радиочастотной метки);

R — устройство считывания/опроса (УСО) (также иногда называемое устройством считывания);

R=>T — направление передачи от УСО к радиочастотной метке;

RTcal — калибровочный символ при передаче R=>T;

T — радиочастотная метка;

$T_1$ — время между запросом УСО и ответом радиочастотной метки;

$T_2$ — время между ответом радиочастотной метки и запросом УСО;

$T_3$ — время ожидания УСО, после T1, до подачи новой команды;

$T_4$ — минимальное время между командами УСО;

$T_f$ or $T_{f,10-90\ \%}$ — время затухания огибающей радиочастотного сигнала;

$T_{pri}$ — период следования импульсов сигнала связи ($T_{pri}$ = 1/LF);

$T_r$ or $T_{r,10-90\ \%}$ — время нарастания огибающей радиочастотного сигнала;

$TR_{ext}$ — выбор — должен ли заголовку предшествовать пилотный сигнал* (для метода ASK) или бит 3 выбора канала для ответа (для метода PJM);

$T_s$ — время установления радиочастотного сигнала;

T=>R — направление передачи от радиочастотной метки к УСО;

---

* *Предварительный тоновый сигнал.*

TRcal — калибровочный символ при передаче T=>R;

$X_{fp}$ — значение с плавающей запятой,

$xxxx_2$ — двоичная система счисления;

$xxxx_h$ — шестнадцатеричная система счисления;

≈ — МОДА 1 — приближенное значение (например, ≈ 75,52 мкс).

### 5.2 Сокращения

ARIB — Ассоциация радио индустрии и бизнеса (Association of Radio Industries and Businesses);

AFI — идентификатор семейства приложений (Application family identifier);

AM — амплитудная модуляция (Amplitude modulation);

ASK — амплитудная манипуляция (Amplitude shift keying);

BPSK — двоичная фазовая манипуляция (Binary Phase Shift Keying);

CEPT — Европейская конференция администраций почтовых служб и служб связи (Conference of European Posts and Telecommunications);

CFR — Кодекс федеральных правил (свод федеральных регулирующих актов США) (Code of Federal Regulations);

CRC — циклический избыточный код (Cyclic Redundancy Check);

Примечание — В настоящем стандарте используются два типа циклического избыточного кода: CRC-5 (5-битовый CRC) и CRC-16 (16-битовый CRC) и три различных понятия CRC-16: код StoredCRC, код PacketCRC и CRC-16c.

Для памяти UII (0-го слова) и команды ACK используются следующие два понятия CRC-16:
- код StoredCRC = CRC-16 рассчитывается при запуске и отображается в 0 слове памяти UII;
- код PacketCRC = CRC-16 рассчитывается по данным ответа радиочастотной метки на команду ACK.

Для всех остальных случаев и команд используется следующее понятие CRC-16:
- CRC-16c = CRC-16 рассчитывается по данным ответа, полученного от радиочастотной метки.

CW — непрерывный радиочастотный сигнал (Continuous wave);

dBch — уровень в децибелах относительно интегральной мощности в опорном канале (Decibels referenced to the integrated power in the reference channel);

DSB — боковые полосы (Double sideband);

DSB-ASK — амплитудная манипуляция с двумя боковыми полосами (Double-SideBand Amplitude-Shift Keying);

DR — отношение (результат деления) (Divide ratio);

ERC — Европейский комитет по радиосвязи (European Radiocommunications Committee);

ERM — электромагнитная совместимость и вопросы радиочастотного спектра (Electromagnetic compatibility and Radio spectrum Matters);

ETSI — Европейский институт стандартизации электросвязи (European Telecommunications Specifications Institute);

FCC — Федеральная комиссия по связи (ФКС) (Federal Communications Commission);

FT — допустимое отклонение частоты (погрешность частоты) (Frequency Tolerance);

ITF — «УСО передает первым» (или «устройство считывания передает первым»)* (Interrogator talks first (reader talks first));

LF — частота сигнала связи (LF = $1/T_{pri}$) (Link Frequency (LF = $1/T_{pri}$));

MFM — модифицированная частотная модуляция (Modified Frequency Modulation);

N/A — не применимо (Not Applicable);

NSI — идентификатор системы счисления (Numbering system identifier);

PIE — время-импульсное кодирование (Pulse-Interval Encoding);

---

* *Протокол ITF — протокол для связи, согласно которому радиочастотная метка передает сигнал ответа только после получения команды УСО.*

PJM — фазовая модуляция колебаний (Phase Jitter Modulation);

ppm — частей на миллион (Parts-per-million);

PC — управление протоколом (Protocol control);

RF — частота радиосигнала (Radio frequency);

RFU — зарезервировано для будущего использования (Reserved for future use);

RN16 — 16-разрядное случайное или псевдослучайное число (16-bit random or pseudo-random number);

RNG — генератор случайных или псевдослучайных чисел (Random or pseudo-random number generator);

SRD — средства радиосвязи малого радиуса действия (Short Range Devices);

TDM — мультиплексирование с разделением по времени или мультиплексировано с разделением по времени (по контексту) (Time-division multiplexing or time-division multiplexed (as appropriate));

TID — идентификация радиочастотной метки или идентификатор радиочастотной метки в зависимости от контекста (Tagidentification or tag identifier, depending on context);

UII — уникальный идентификатор предмета учета (Unique item identifier);

UMI — индикатор пользовательской памяти (User-memory indicator);

XI — индикатор XPC (XPC indicator);

XPC — расширенное управление протоколом (Extended protocol control);

XPC_W1 — слово 1 XPC (XPC word 1);

XPC_W2 — слово 2 XPC (XPC word 2);

XTID — расширенный идентификатор радиочастотной метки (*см. [2]*) (Extended TID indicator (*см. [2]*).

### 5.3 Примечания

При описании МОДА 3 настоящей спецификации приняты следующие обозначения:

- состояния и флаги состояний выделены полужирным. Пример: **ready**;
- команды обозначены курсивом. Переменные также обозначаются курсивом. Там, где в спецификации может возникнуть путаница между обозначениями команд и переменных, должно быть явное указание в самом наименовании. Пример: *BeginRound* (Начать цикл опроса);
- процедуры показаны ***подчеркнутым курсивом***;
- параметры команд подчеркнуты. Пример: Pointer (Указатель);
- для указания на логическое отрицание, перед обозначением используется символ '~'. Пример: Если значение «**flag**» – истина, то «**~ flag**» – ложь;
- обозначение «R=>T» относится к командам или к сигналам связи, передаваемым по радиоинтерфейсу от УСО к радиочастотной метке (англ. reader-to-tag));
- обозначение «T=>R» относится к командам или к сигналам связи, передаваемым по радиоинтерфейсу от радиочастотной метки к УСО (англ. tag-to-reader).

## 6 Требования: физический уровень, система управления коллизиями и протокол передачи данных для систем, работающих на частоте 13,56 МГц

### 6.0 Общие положения, применимые ко всем МОДА

#### 6.0.1 Представление в соответствии с *[1]*

Контекст, формулировки и представления настоящего стандарта, описывающие физический уровень, систему управления коллизиями и протокол передачи данных для систем радиочастотной идентификации, работающих в диапазоне 13,56 МГц, находятся в полном соответствии с требованиями *[1]*.

#### 6.0.2 Функциональная совместимость с настоящим стандартом

Настоящий стандарт определяет и описывает три МОДА на частоте 13,56 МГц.

Описанные МОДА не совместимы между собой, но полагается, что при работе они не оказывают какого-либо существенного воздействия друг на друга. Все известные перекрестные взаимодействия перечислены в приложении О.

Примечания

1 Рекомендуется, чтобы пользователи выбирали одну из МОДА для каждого конкретного применения.

2 В соответствии с национальными требованиями использования радиочастотного спектра могут существовать ограничения по мощности, частоте или по полосе пропускания радиочастотного спектра, что снижает производительность системы при применении ее в данной стране. Это означает, что пользователь несет ответственность за получение подтверждения от изготовителей, а также, при необходимости, за проведение соответствующих испытаний, чтобы гарантировать соответствие системы требованиям законодательства.

3 На момент подготовки настоящего стандарта канал связи «УСО-радиочастотная метка» и «радиочастотная метка-УСО» может быть объектом для получения официального разрешения или проведения сертификации, поэтому, в дополнение к настоящему стандарту, необходимо сделать ссылку на национальное регулирование и стандарты в области радиосвязи. Все системы должны быть согласованы с национальными требованиями использования радиочастотного спектра.

**6.0.3 Соответствие устройства опроса/считывания требованиям настоящего стандарта**

Для утверждения о соответствии устройства опроса/считывания настоящему стандарту УСО должен поддерживать: МОДА 1, МОДА 2 либо МОДА 3. УСО может поддерживать любой или все МОДА как опцию (МОДА не являются взаимодействующими).

**6.0.4 Соответствие радиочастотной метки требованиям настоящего стандарта**

Для утверждения о соответствии радиочастотной метки настоящему стандарту радиочастотная метка должна поддерживать: МОДА 1, МОДА 2 либо МОДА 3. Радиочастотная метка может поддерживать любую или все МОДА как опцию (режимы не являются взаимодействующими).

**6.0.5 Структура команд и возможность расширения**

6.1, 6.2 и 6.3 описывают структуру кодов команд между УСО и радиочастотной меткой и показывают, какое количество позиций доступно для будущего расширения. Подразделы, описывающие спецификации команд управления, полностью описывают команды и их представление. Каждая команда помечается как «обязательная» или «опциональная». В соответствии с *[1]* подразделы настоящего стандарта устанавливают возможность реализации «команд пользователя» и «команд изготовителя».

Виды характеристик разрешенных команд определены в пунктах от 6.0.6 до 6.0.9.

**6.0.6 Обязательные команды**

Обязательная команда должна поддерживаться всеми радиочастотными метками, отвечающими требованиям стандарта. УСО, отвечающие требованиям стандарта, должны поддерживать все обязательные команды.

**6.0.7 Опциональные команды**

Опциональные команды — команды, которые определены в настоящем стандарте. УСО должны быть технически способными к выполнению всех опциональных команд, которые определены в настоящем стандарте (хотя необходимости быть сконфигурированными для этого нет). Радиочастотные метки могут поддерживать или не поддерживать опциональные команды.

Если в устройстве опроса или в радиочастотной метке реализована опциональная команда, то она должна соответствовать требованиям настоящего стандарта.

**6.0.8 Команды пользователя**

Команды пользователя могут быть разрешены к использованию настоящим стандартом, но не должны быть описаны в настоящем сандарте.

Команда пользователя не должна никаким образом копировать функциональность какой-либо обязательной или дополнительной команды, определенной в настоящем стандарте. УСО должен использовать команду пользователя только в соответствии со спецификациями изготовителя радиочастотной метки.

**6.0.9 Команды изготовителя**

Команды изготовителя могут быть разрешены к использованию настоящим стандартом, но не должны быть описаны в настоящем стандарте.

Команда изготовителя не должна никаким образом копировать функциональность какой-либо обязательной или дополнительной команды, определенной в настоящем стандарте. Поставщики не должны предоставлять каких-либо средств для обхода ограничений, накладываемых протоколом. Команды изготовителя предназначены для производственных целей и не должны использоваться в установленных системах радиочастотной идентификации.

### 6.1 МОДА 1: физический уровень, система управления коллизиями и протоколы МОДА 1

МОДА 1 не совместима ни с какими другими МОДА, определенными в настоящем стандарте.

**6.1.1 Система считывания/записи**

МОДА 1 описывает систему считывания/записи, используя метод «Обмен данными начинает УСО».

**6.1.2 Нормативные аспекты**

Физический уровень, система управления коллизиями и протоколы передачи, описанные в этой МОДА, определены в *ГОСТ Р ИСО/МЭК 15693-1*, *ГОСТ Р ИСО/МЭК 15693-2*. Пункты 6.1.3—6.1.8 описывают нормативные требования МОДА 1.

**6.1.3 Аспекты измерения параметров соответствия и производительности**

Аспекты измерения параметров соответствия и производительности для МОДА 1 даны в соответствующих подразделах (*ГОСТ Р ИСО/МЭК 18046* и *ГОСТ Р 56914* соответственно).

**6.1.4 Физический уровень**

Физический уровень МОДА 1 радиоинтерфейса на частоте 13,56 МГц должен соответствовать требованиям *ГОСТ Р ИСО/МЭК 15693-2*.

**6.1.5 Протокол и метод управления коллизиями**

Протокол и метод управления коллизиями МОДА 1 радиоинтерфейса на частоте 13,56 МГц должен соответствовать требованиям *ГОСТ Р ИСО/МЭК 15693-3*.

**6.1.6 Команды**

Команды МОДА 1 для радиоинтерфейса на частоте 13,56 МГц должны соответствовать требованиям *ГОСТ Р ИСО/МЭК 15693-3*. В приложении А приведены команды, которые необходимы, чтобы поддерживать кодирование, описанное в *[3]*, и которые требуются в радиочастотной идентификации для управления предметами.

**6.1.7 Таблица параметров канала связи «УСО-радиочастотная метка»**

Таблица параметров канала связи «УСО-радиочастотная метка» для МОДА 1 радиоинтерфейса на частоте 13,56 МГц, должна соответствовать требованиям *ГОСТ Р ИСО/МЭК 15693-2*. Для получения более подробной информации см. таблицу 1.

Таблица 1 — Параметры канала связи «УСО-радиочастотная метка»

| Ссылка | Параметр | Описание/граничные значения | Опции/замечания |
|---|---|---|---|
| M1-Int: 1 | Рабочий диапазон частот | 1 канал связи «УСО-радиочастотная метка» на частоте (центральная частота) 13,56 Мгц ± 7 кГц | — |
| M1-Int: 1a | Рабочая частота по умолчанию | 13,56 Мгц | — |
| M1-Int: 1b | Рабочие каналы (для систем с расширенным спектром) | N/A | — |
| M1-Int: 1c | Точность рабочей частоты | +/–100 миллионных долей<br>+/–50 миллионных долей в Японии | — |
| M1-Int: 1d | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A | — |
| M1-Int: 1e | Последовательность значений рабочих частот (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A | — |
| M1-Int: 2 | Полоса пропускания используемого канала | 13,56 МГц ± 7 кГц с модуляцией согласно 6.1.4 | — |
| M1-Int: 2a | Минимальный частотный диапазон приемника | 13,56 МГц ± (423,75 кГц ± 40 кГц)<br>13,56 МГц ± (484,28 кГц ± 40 кГц) | Центрируемый на поднесущей частоте |

*Продолжение таблицы 1*

| Ссылка | Параметр | Описание/граничные значения | Опции/замечания |
|---|---|---|---|
| M 1-Int: 3 | Максимальная напряженность магнитного поля УСО.<br>Предельные значения напряженности магнитного поля в коммуникационной зоне | УСО не должно создавать поле напряженностью более 12 А/м в любой части объема, где может находится радиочастотная метка, размер которой соответствует размеру идентификационной карты по *[4]*. Максимальная рабочая напряженность поля: 5 А/м для радиочастотных меток, размер которых соответствует размеру идентификационной карты по *[4]*. Для других типоразмеров меток изготовитель должен указать максимальную рабочую напряженность поля* | (Индуктивная связь)<br>Методы испытаний для определения рабочей области УСО определены в *ГОСТ Р 56914* |
| M1-Int: 3a | Минимальная рабочая напряженность поля | Минимальная рабочая напряженность поля: 150 мА/м для радиочастотных меток, размер которых соответствует размеру идентификационной карты по *[4]*. В различных применениях могут использоваться разные значения минимальной рабочей напряженности поля. В этом случае минимальная рабочая напряженность поля определяется изготовителем радиочастотной метки | — |
| M1-Int: 4 | Паразитное излучение УСО | В соответствии с национальными требованиями использования радиочастотного спектра** | — |
| M1-Int: 4a | Паразитное излучение УСО внутри рабочего диапазона частот (для систем с расширенным спектром) | N/A | — |
| M1-Int: 4b | Паразитное излучение УСО вне рабочего диапазона частот | N/A | — |
| M1-Int: 5 | Спектральная маска передатчика УСО | Метод модуляции и поразрядное кодирование допускают максимальное значение мощности поля для метки в соответствии со следующими нормами:<br>США: FCC 47 часть 15<br>Евросоюз: EN 300—330<br>Япония: ARIB STD—T82 | — |
| M1-Int:6 | Временные диаграммы | См. ниже | — |
| M1-Int: 6a | Время цикла передача — прием | <=4320/fc | — |
| M1-Int: 6b | Время цикла прием — передача | >=4192/fc | — |
| M1-Int: 6c | Время задержки или линейно нарастающий сигнал УСО | См. 6.1.4 | — |
| M1-Int: 6d | Время затухания или линейно падающий сигнал УСО | См. 6.1.4 | — |

* *Решением ГКРЧ 07-20-03-001 определена максимальная напряженность магнитного поля на расстоянии 10 м от УСО, которая составляет 60 дБ (мкА/м).*

** *В соответствии с решением ГКРЧ от 7 мая 2007 года № 07-20-03-001, г. Москва, «О выделении полос радиочастот устройствам малого радиуса действия» для устройств радиочастотной идентификации» установлена максимальная напряженность магнитного поля на расстоянии 10 м от УСО на уровне 60 дБ (мкА/м).*

*Окончание таблицы 1*

| Ссылка | Параметр | Описание/граничные значения | Опции/замечания |
|---|---|---|---|
| M1-Int: 7 | Модуляция | Амплитудная модуляция несущей частоты (АМн 100 %, АМн 10 %) | — |
| M1-Int: 7a | Последовательность значений частоты (для систем псевдослучайной перестройки частоты) | N/A | — |
| M1-Int: 7b | Тактовая частота (для систем псевдослучайной перестройки частоты) | N/A | — |
| M1-Int: 7c | Точность значений тактовой частоты (для систем псевдослучайной перестройки частоты) | N/A | — |
| M1-Int: 7d | Индекс модуляции | Два уровня амплитудной модуляции: 100 % и 10 %.<br>Индекс модуляции: (a – b)/(a + b).<br>Примечание — Глубина модуляции (a – b)/a | Любой из этих индексов определяется УСО (см. 6.1.4). Два индекса модуляции допустимы и определяются УСО. Оба должны быть декодированы радиочастотной меткой. Модуляция должна использовать принцип ASK (амплитудной манипуляции). В зависимости от выбора УСО |
| M1-Int: 7e | Рабочий цикл | См. 6.1.4 | — |
| M1-Int: 7f | Девиация частотной модуляции | N/A | — |
| M1-Int: 8 | Кодирование данных | Кодирование данных должно быть реализовано, с использованием фазово-импульсной модуляции. Радиочастотной меткой должны поддерживаться два режима кодирования данных. Выбор режима должен быть сделан УСО и указан радиочастотной метке в стартовом кадре (SOF) | Режим кодирования данных «1 из 256»:<br>Значение одного байта должно быть представлено позицией одного импульса. Позиция импульса на одном из 256 следующих друг за другом временных интервалов с периодом 18,88 мкс (256/Fc) определяет значение байта.<br>Режим кодирования данных «1 из 4»:<br>Фазово-импульсная модуляция должна использоваться там, где позиция определяет два бита за один раз. Четыре последовательные пары битов формируют байт.<br>Результирующая скорость передачи данных в битах составляет 26,48 Кбит/с (fc/512) |
| M1-Int: 9 | Скорость передачи данных в битах | Режим кодирования «1 из 256»: 1,65 Кбит/с (fc/8192).<br>Режим кодирования «1 из 4»: 26,48 Кбит/с (fc/512) | Любой режим, определяется УСО |
| M1-Int: 9 | Скорость передачи данных в битах | Режим кодирования «1 из 256»: 1,65 Кбит/с (fc/8192).<br>Режим кодирования «1 из 4»: 26,48 Кбит/с (fc/512) | Любой режим, определяется УСО |
| M1-Int: 9a | Точность скорости передачи данных в битах | Соответствует точности несущей частоты | — |
| M1-Int: 10 | Точность модуляции УСО | N/A | — |

### 6.1.8 Таблица параметров для канала связи «радиочастотная метка-УСО»

Таблица параметров канала связи «радиочастотная метка-УСО» для МОДА 1 радиоинтерфейса на частоте 13,56 МГц должна соответствовать требованиям *ГОСТ Р ИСО/МЭК 15693-2*. Для получения более подробной информации см. таблицу 2.

Таблица 2 — Параметры канала связи «радиочастотная метка-УСО»*

| Позиция | Параметр | Описание/граничные значения | Опции/ замечания |
|---|---|---|---|
| M1-Tag: 1 | Рабочий диапазон частот. Диапазоны поднесущих частот | Могут использоваться одна или две поднесущие. Выбор между ними должен быть сделан УСО, используя первый бит в заголовке протокола, как определено в *ГОСТ Р ИСО/МЭК 15693-3*.<br>Когда используется одна поднесущая, частота fs1 поднесущей модуляции нагрузкой должна быть fc/32 (423,75 кГц). Когда используются две поднесущие, частота fs1 должна быть fc/32 (423,75 кГц), а частота fs2 должна быть fc/28 (484,28 кГц).<br>Когда используются две поднесущие, фаза должна быть непрерывной, как показано на диаграммах | — |
| M1-Tag: 1a | Рабочая частота по умолчанию | 13,56 МГц ± 7 кГц | — |
| M1-Tag: 1b | Рабочие каналы (для систем с расширенным спектром) | N/A | — |
| M1-Tag: 1c | Точность рабочей частоты | Точность значения частоты 13,56 МГц соответствует точности несущей частоты | — |
| M1-Tag: 1d | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A | — |
| M1-Tag: 1e | Последовательность значений рабочих частот (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A | — |
| M1-Tag: 2 | Полоса пропускания используемого канала | 13,56 МГц: 4 поднесущие<br>1 однесущая: 13,56 МГц ± (423,75 кГц ± 40 кГц)<br>2 поднесущие: 2 канала: 13,56 МГц ± (484,28 кГц ± ± 40 кГц) | — |
| M1-Tag: 3 | Максимальная напряженность магнитного поля | В соответствии с национальными требованиями использования радиочастотного спектра* | — |
| M1-Tag: 4 | Паразитное излучение | N/A | — |
| M1-Tag: 4a | Паразитное излучение внутри рабочего диапазона частот (для систем с расширенным спектром) | N/A | — |
| M1-Tag: 4b | Паразитное излучение вне рабочего диапазона частот | N/A | — |
| M1-Tag: 5 | Спектральная маска передатчика | В соответствии с:<br>США: FCC 47 часть 15<br>ЕС и другие EN:300—330 (Спектральная маска передатчика не регулируется законом Японии о радио)** | — |

* *В соответствии с решением ГКРЧ от 7 мая 2007 года № 07-20-03-001, г. Москва, «О выделении полос радиочастот устройствам малого радиуса действия» для устройств радиочастотной идентификации» установлена максимальная напряженность магнитного поля на расстоянии 10 м от УСО на уровне 60 дБ(мкА/м).*

** *В Российской Федерации не регулируется.*

*Продолжение таблицы 2*

| Позиция | Параметр | Описание/граничные значения | Опции/ замечания |
|---|---|---|---|
| M1-Tag: 6 | Временные диаграммы | См. ниже. | — |
| M1-Tag: 6a | Время цикла передача—прием | <=4192/fc | — |
| M1-Tag: 6b | Время цикла прием—передача | От 4320/fc до 4384/fc | — |
| M1-Tag: 6c | Время задержки или линейно нарастающий сигнал | N/A | — |
| M1-Tag: 6d | Время затухания или линейно падающий сигнал | N/A | — |
| M1-Tag: 7 | Модуляция (на несущей) | Должна быть возможность связи радиочастотной метки с УСО в среде, поддерживающей индуктивное соединение, посредством модуляции несущей частоты, обеспечивающей генерацию поднесущей на частоте fs.<br>Поднесущая должна генерироваться путем изменения нагрузки в радиочастотной метке.<br>Методы испытаний модуляции нагрузкой радиочастотной метки определены в *ГОСТ Р 56914*.<br>Амплитуда модуляции нагрузкой должна быть не менее 10 мВ для радиочастотных меток, размер которых соответствует размеру идентификационной карты по *[4]* при измерении в соответствии с описанными методами испытаний.<br>Для радиочастотных меток других форм-факторов, изготовитель должен определить параметры модуляции нагрузкой | — |
| M1-Tag: 7a | Последовательность значений частоты (для систем с псевдослучайной перестройкой частоты [FHSS]) | N/A | — |
| M1-Tag: 7b | Тактовая частота (для систем на основе сигналов с расширенным спектром) | N/A | — |
| M1-Tag: 7c | Точность значений тактовой частоты (для систем на основе сигналов с расширенным спектром) | N/A | — |
| M1-Tag: 7d | Отношение ON/OFF* | N/A | — |
| M1-Tag: 7e | Частота поднесущей | Если используется одна поднесущая: 423,75 кГц.<br>Если используются две поднесущие: 423,75 кГц и 484,28 кГц | — |
| M1-Tag: 7f | Точность частоты поднесущей. Допустимое отклонение частоты поднесущей, сгенерированной в канале связи «радиочастотная метка-УСО) | Определяется точностью несущей | — |

* *Отношение времени излучения к времени ожидания.*

*Продолжение таблицы 2*

| Позиция | Параметр | Описание/граничные значения | Опции/ замечания |
|---|---|---|---|
| M1-Tag: 7g | Модуляция поднесущей | Одна поднесущая: логический **0** передается 8 импульсами на частоте 423,75 кГц (fc/32), за которыми следует временной интервал 18,88 мкс (256/fc), в течение которого модуляция отсутствует.<br>Логическая **1** начинается временным интервалом 18,88 мкс (256/fc), в течение которого модуляция отсутствует, за которым следуют 8 импульсов на частоте 423,75 кГц (fc/32).<br>Две поднесущие: Логический **0** передается 8 импульсами на частоте 423,75 кГц (fc/32), за которыми следуют 9 импульсов на частоте 484,28 кГц (fc/28).<br>Логическая **1** передается 9 импульсами на частоте 484,28 кГц (fc/28), за которыми следуют 8 импульсов на частоте 423,75 кГц (fc/32) | Определяется системой опроса. Радиочастотная метка должна поддерживать обе |
| M1-Tag: 7h | Рабочий цикл | N/A | — |
| M1-Tag: 7 I | Девиация частотной модуляции | N/A | — |
| M1-Tag: 8 | Кодирование данных | Данные должны быть закодированы, используя Манчестерское кодирование одной или двух поднесущих | — |
| M1-Tag: 9 | Скорость передачи данных в битах | См. 6.1.4 | — |
| M1-Tag: 9a | Точность скорости передачи данных в битах | Определяется точностью несущей частоты | — |
| M1-Tag: 10 | Точность модуляции (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A | — |
| M1-Tag: 11 | Заголовок | Кадрирование было выбрано для простоты синхронизации и независимости протокола. Кадры разделяются на стартовый кадр (SOF) и завершающий кадр (EOF), образованные нарушением кодирования | — |
| M1-Tag: 11a | Длина заголовка | См. 6.1.4 | — |
| M1-Tag: 11b | Форма сигнала заголовка | **: SOF при использовании одной поднесущей**<br>SOF состоит из 3 частей:<br>1: времени отсутствия модуляции 56,64 мкс (768/fc);<br>2: 24-х импульсов на частоте 423,75 кГц (fc/32);<br>3: последующей логической **1**, которая передается интервалом времени 18,88 мкс (256/fc), в течение которого модуляция отсутствует, за которым следуют 8 импульсов на частоте 423,75 кГц (fc/32).<br>**: SOF при использовании двух поднесущих**<br>SOF состоит из 3 частей:<br>1: 27-и импульсов на частоте 484,28 кГц (fc/28);<br>2: 24-х импульсов на частоте 423,75 кГц (fc/32);<br>3: логической **1**, которая передается 9 импульсами на частоте 484,28 кГц (fc/28), за которыми следуют 8 импульсов на частоте 423,75 кГц (fc/32)<br>**: EOF при использовании одной поднесущей**<br>EOF состоит из 3 частей:<br>1: логического **0**, который передается 8 импульсами на частоте 423,75 кГц (fc/32); | — |

*Окончание таблицы 2*

| Позиция | Параметр | Описание/граничные значения | Опции/ замечания |
|---|---|---|---|
| | | 2: интервала времени 18,88 мкс (256/fc), в течение которого модуляция отсутствует;<br>3: 24 импульсов на частоте 423,75 кГц (fc/32), интервала времени 56,64 мкс (768/fc), в течение которого модуляция отсутствует<br>: **EOF при использовании двух поднесущих**<br>EOF состоит из 3 частей:<br>1: логического **0**, который передается 8 импульсами на частоте 423,75 кГц (fc/32), за которыми следуют 9 импульсов на частоте 484,28 кГц (fc/28);<br>2: 24 импульсов на частоте 423,75 кГц (fc/32);<br>3: 27 импульсов на частоте 484,28 кГц (fc/28) | |
| M1-Tag: 11c | Последовательность битов синхронизации | N/A | — |
| M1-Tag: 12 | Скремблирование (для систем на основе сигналов с расширенным спектром) | N/A | — |
| M1-Tag: 13 | Порядок передачи битов | Начиная с младшего значащего бита | — |
| M1-Tag: 14 | Резервный | | — |
| M1-Tag: 15 | Поляризация | N/A (поле в ближней зоне) | — |
| M1-Tag: 16 | Минимальная полоса пропускания приемника радиочастотной метки | См. передаваемые/принимаемые сигналы 6.1.4 | — |

### 6.2 МОДА 2: физический уровень, система управления коллизиями и протоколы МОДА 2

МОДА 2 не совместим ни с какими другими режимами МОДА, определенными в настоящем стандарте.

МОДА 2 не влияет на любой другой режим, определенный в настоящем стандарте.

Аспекты измерения параметров соответствия и производительности для МОДА 2 должны удовлетворять *ГОСТ Р ИСО/МЭК 18046* и *ГОСТ Р 56914* соответственно.

#### 6.2.1 Нормативные аспекты: физические параметры и параметры управления доступом к среде (MAC): канал связи «УСО-радиочастотная метка»

Физические параметры и параметры управления доступом к среде (MAC) канала связи «УСО-радиочастотная метка» приведены в таблице 3.

Таблица 3 — Физические параметры и параметры управления доступом к среде (MAC) канала связи: «УСО-радиочастотная метка»

| Позиция | Параметр | Описание |
|---|---|---|
| M2-Int: 1 | Диапазон рабочих частот | 13,56 МГц ± 7 кГц |
| M2-Int:1a | Рабочая частота по умолчанию | 13,56 МГц |
| M2-Int: 1b | Рабочие каналы (для систем на основе сигналов с расширенным спектром) | N/A |
| M2-Int: 1c | Точность рабочей частоты | ±100 миллионных долей<br>±50 миллионных долей в Японии |
| M2-Int: 1d | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки частоты [FHSS]) | N/A |

*Продолжение таблицы 3*

| Позиция | Параметр | Описание |
|---|---|---|
| M2-Int: 1e | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A |
| M2-Int: 2 | Полоса пропускания используемого канала | Модуляция боковых полос частот мала по амплитуде, но имеет широкий частотный спектр. Они удовлетворяют нормам ETSI и FCC. Полоса пропускания используемого канала не регулируется Законом Японии о Радио |
| M2-Int: 2a | Минимальная полоса пропускания приемника | Соответствует каналу, необходимому для приема данных, или рассматриваемому каналу |
| M2-Int: 3 | Максимальная напряженность магнитного поля УСО | УСО не должен создавать поле напряженностью более 12 А/м в любой области пространства, в которой может находиться радиочастотная метка, размер которой соответствует размеру идентификационной карты по ИСО* *[4]* |
| M2-Int: 3a | Минимальная рабочая напряженность поля | Минимальная рабочая напряженность поля: 150 мА/м для радиочастотных меток, размер которых соответствует размеру идентификационной карты по ИСО.<br>В различных приложениях к значению минимальной рабочей напряженности поля могут быть предъявлены иные требования. В этом случае необходимо ориентироваться на значение минимальной рабочей напряженности поля, указанной изготовителем радиочастотной метки |
| M2-Int: 4 | Паразитное излучение УСО | См. ниже |
| M2-Int: 4a | Паразитное излучение внутри рабочего диапазона частот (для систем с расширенным спектром) | N/A |
| M2-Int: 4b | Паразитное излучение вне рабочего диапазона частот | Соответствует ETSI, ARIB STD-T82 и FCC по максимально разрешенной напряженности поля вне рабочего диапазона частот |
| M2-Int: 5 | Спектральная маска передатчика УСО | Соответствует ETSI, ARIB STD-T82 и FCC по максимально разрешенной напряженности поля вне рабочего диапазона частот |
| M2-Int: 6 | Временные диаграммы | См. ниже |
| M2-Int: 6a | Время цикла передача—прием (время между окончанием команды, и готовностью УСО получить ответ) | От 0 мкс до 50 мкс |
| M2-Int: 6b | Время цикла прием—передача (время между окончанием ответа и готовностью УСО передать новую команду) | Больше, чем 0 мкс |
| M2-Int: 6c | Время задержки или линейно нарастающий сигнал УСО | От 0 мкс до 10 мкс |
| M2-Int: 6d | Время затухания или линейно падающий сигнал | От 0 мкс до 10 мкс |

** Решением ГКРЧ 07-20-03-001 определена максимальная напряженность магнитного поля на расстоянии 10 м от УСО, которая составляет 60 дБ(мкА/м).*

*Окончание таблицы 3*

| Позиция | Параметр | Описание |
|---|---|---|
| M2-Int: 7 | Модуляция | PJM (фазовая модуляция колебаний) мин. уровень ±3,0°, макс. уровень ±4,0° |
| M2-Int: 7a | Последовательность значений частоты (для систем псевдослучайной перестройки частоты [FHSS]) | N/A |
| M2-Int: 7b | Тактовая частота (для систем на основе сигналов с расширенным спектром) | N/A |
| M2-Int: 7c | Точность значений тактовой частоты (для систем на основе сигналов с расширенным спектром) | N/A |
| M2-Int: 7d | Коэффициент модуляции | N/A (Система не использует амплитудную модуляцию) |
| M2-Int: 7e | Рабочий цикл | N/A |
| M2-Int: 7f | Девиация частотной модуляции | N/A |
| M2-Int: 8 | Кодирование данных | Модифицированная частотная модуляция (MFM-кодирование) (см. рисунок 4) |
| M2-Int: 9 | Скорость передачи данных в битах | 423,75 Кбит/с |
| M2-Int: 9a | Точность скорости передачи данных в битах | Определяется точностью несущей частоты |
| M2-Int: 10 | Точность модуляции передатчика УСО | N/A |
| M2-Int: 11 | Заголовок | Включает изменение метода модифицированной частотной модуляции |
| M2-Int: 11a | Длина заголовка | 16 бит |
| M2-Int: 11b | Форма сигнала заголовка | Флаг команды определяет начало команды и длительность бита. Флаг состоит из трех частей:<br>1: синхронизирующей последовательности из 8 бит действительных данных MFM.<br>2: В обычных данных MFM нет нарушений кодирования модифицированной частотной модуляции (MFM). Нарушение представляет собой последовательность из 5 изменений состояния, разделенных соответственно: интервалом в 1 бит, интервалом в 2 бита, интервалом в 1,5 бита и интервалом в 2 бита. Пятый (последний) переход определяет начало битового интервала.<br>3: Завершающий MFM 0 определяет конец флага и начало команды.<br>(См. рисунок 1) |
| M2-Int: 11c | Последовательность битов синхронизации | См. M2 Int: 11b |
| M2-Int: 11d | Последовательность синхронизации кадра | См. M2 Int: 11b |
| M2-Int: 12 | Скремблирование (для систем с расширенным спектром) | N/A |
| M2-Int: 13 | Порядок передачи битов | Начиная с младшего значащего бита |
| M2-Int: 14 | Процесс пробуждения | «УСО передает первым». Радиочастотная метка не может ответить, пока не получит действительную команду от УСО |
| M2-Int: 15 | Поляризация | N/A |

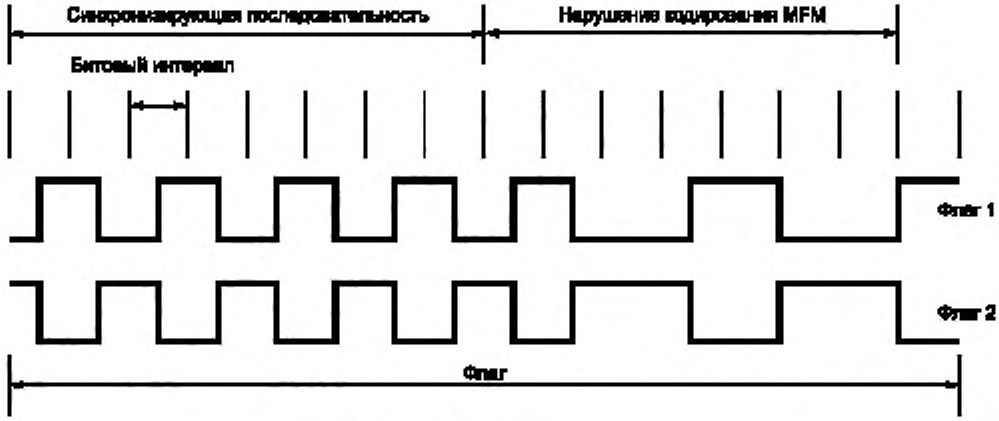



Рисунок 1 — Два возможных флага команд

### 6.2.2 Канал связи «радиочастотная метка-УСО»

Параметры канала связи «радиочастотная метка-УСО» приведены в таблице 4, два возможных флага ответа изображены на рисунке 2.

Таблица 4 — Канал связи «радиочастотная метка-УСО»

| Позиция | Параметр | Описание |
|---|---|---|
| M2-Tag:1 | Диапазон рабочих частот | 13,56 МГц ± 3,013 МГц |
| M2-Tag: 1a | Рабочая частота по умолчанию | N/A (Система не работает на рабочей частоте по умолчанию) |
| M2-Tag: 1b | Рабочие каналы (для систем на основе сигналов с расширенным спектром) | Многочастотная операционная система, где радиочастотные метки могут выбрать один из 8 каналов ответа. Радиочастотные метки передают ответ полностью, используя выбранный канал. Радиочастотная метка может использовать одну из восьми поднесущих. Поднесущие получены путем деления частоты силового питающего поля УСО |
| M2-Tag: 1c | Точность рабочей частоты | Соответствует точности несущей частоты |
| M2-Tag: 1d | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки частоты [FHSS]) | Радиочастотные метки передают ответ полностью, используя выбранный канал |
| M2-Tag: 1e | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | Канал ответа выбирается радиочастотной меткой случайным образом |

| Канал | Частота *поднесущей*, кГц | Коэффициент деления |
|---|---|---|
| A | 969 | 14 |
| B | 1233 | 11 |
| C | 1507 | 9 |
| D | 1808 | 7,5 |
| E | 2086 | 6,5 |
| F | 2465 | 5,5 |
| G | 2712 | 5 |
| H | 3013 | 4,5 |

*Продолжение таблицы 4*

<table>
<tr><th>Позиция</th><th>Параметр</th><th>Описание</th></tr>
<tr><td>M2-Tag: 2</td><td>Полоса пропускания используемого канала</td><td>106 кГц для каждого из 8 каналов ответа</td></tr>
<tr><td>M2-Tag: 3</td><td>Максимальная напряженность магнитного поля УСО</td><td>В соответствии с российскими правилами использования радиочастотного спектра</td></tr>
<tr><td>M2-Tag: 4</td><td>Паразитное излучение</td><td>N/A</td></tr>
<tr><td>M2-Tag: 4a</td><td>Паразитное излучение внутри рабочего диапазона частот (для систем на основе сигналов с расширенным спектром)</td><td>N/A</td></tr>
<tr><td>M2-Tag: 4b</td><td>Паразитное излучение вне рабочего диапазона частот</td><td>N/A</td></tr>
<tr><td>M2-Tag: 5</td><td>Спектральная маска передатчика</td><td>N/A</td></tr>
<tr><td>M2-Tag: 6</td><td>Временные диаграммы</td><td>См. ниже</td></tr>
<tr><td>M2-Tag: 6a</td><td>Время цикла передача—прием (время между окончанием ответа и готовностью радиочастотной метки получить следующую команду)</td><td>От 0 мкс до 200 мкс</td></tr>
<tr><td>M2-Tag: 6b</td><td>Время цикла прием—передача (время между окончанием последнего полученного бита команды и началом первого бита ответа радиочастотной метки)</td><td>От 77 мкс до 88 мкс</td></tr>
<tr><td>M2-Tag: 6c</td><td>Время задержки или линейно нарастающий сигнал</td><td>N/A</td></tr>
<tr><td>M2-Tag: 6d</td><td>Время затухания или линейно падающий сигнал</td><td>N/A</td></tr>
<tr><td>M2-Tag: 7</td><td>Модуляция</td><td>Модуляция нагрузкой</td></tr>
<tr><td>M2-Tag: 7a</td><td>Последовательность значений частоты (для систем псевдослучайной перестройки частоты [FHSS])</td><td>Радиочастотная метка передает ответ по случайно выбранному каналу или по каналу, выбранному УСО</td></tr>
<tr><td>M2-Tag: 7b</td><td>Тактовая частота (для систем на основе сигналов с расширенным спектром)</td><td>N/A</td></tr>
<tr><td>M2-Tag: 7c</td><td>Точность значений тактовой частоты (для систем на основе сигналов с расширенным спектром)</td><td>N/A</td></tr>
<tr><td>M2-Tag: 7d</td><td>Отношение ON/OFF</td><td>N/A</td></tr>
<tr><td>M2-Tag: 7e</td><td>Частота поднесущих</td><td>8 доступных поднесущих частот:
<table>
<tr><th>Канал</th><th>Частота поднесущей, кГц</th><th>Коэффициент деления</th></tr>
<tr><td>A</td><td>969</td><td>14</td></tr>
<tr><td>B</td><td>1233</td><td>11</td></tr>
<tr><td>C</td><td>1507</td><td>9</td></tr>
<tr><td>D</td><td>1808</td><td>7,5</td></tr>
<tr><td>E</td><td>2086</td><td>6,5</td></tr>
<tr><td>F</td><td>2465</td><td>5,5</td></tr>
<tr><td>G</td><td>2712</td><td>5</td></tr>
<tr><td>H</td><td>3013</td><td>4,5</td></tr>
</table></td></tr>
</table>

*Окончание таблицы 4*

| Позиция | Параметр | Описание |
|---|---|---|
| M2-Tag: 7f | Точность частоты поднесущей | Соответствует точности несущей частоты |
| M2-Tag: 7g | Модуляция поднесущей | BPSK (двоичная фазовая манипуляция) |
| M2-Tag: 7h | Рабочий цикл | N/A |
| M2-Tag: 7I | Девиация частотной модуляции | N/A |
| M2-Tag: 8 | Кодирование данных | MFM (модифицированная частотная модуляция) (см. рисунок 7) |
| M2-Tag: 9 | Скорость передачи данных в битах | 105,9375 Кбит/с |
| M2-Tag: 9a | Точность скорости передачи данных в битах | Соответствует точности несущей частоты |
| M2-Tag: 10 | Точность модуляции передатчика радиочастотной метки (для систем, использующих алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) | N/A |
| M2-Tag: 11 | Заголовок | Включает изменение метода модифицированной частотной модуляции |
| M2-Tag: 11a | Длина заголовка | 16 бит |
| M2-Tag: 11b | Форма сигнала заголовка | Флаг ответа определяет начало ответа и длительность бита. Флаг состоит из трех частей:<br>1: синхронизирующей последовательности из 9-и бит действительных данных MFM.<br>2: В обычных данных нет нарушения кодирования модифицированной частотной модуляции (MFM). Нарушение представляет собой последовательность из 4 изменений состояния, разделенных соответственно: интервалом в 2 бита, интервалом в 1,5 бита, интервалом в 2 бита. Четвертый (последний) переход определяет начало битового интервала.<br>3: Завершающий **0** определяет конец флага. (См. рисунок 2) |
| M2-Tag: 11c | Последовательность битов синхронизации | См. M2-Tag:11b |
| M2-Tag: 11d | Последовательность синхронизации кадра | См. M2-Tag:11b |
| M2-Tag: 12 | Скремблирование (для систем на основе сигналов с расширенным спектром) | N/A |
| M2-Tag: 13 | Порядок передачи битов | Младший значащий бит в начале |
| M2-Tag: 14 | Резервный | — |
| M2-Tag: 15 | Поляризация | N/A |
| M2-Tag: 16 | Минимальная полоса пропускания приемника радиочастотной метки | См. рисунок 3 |

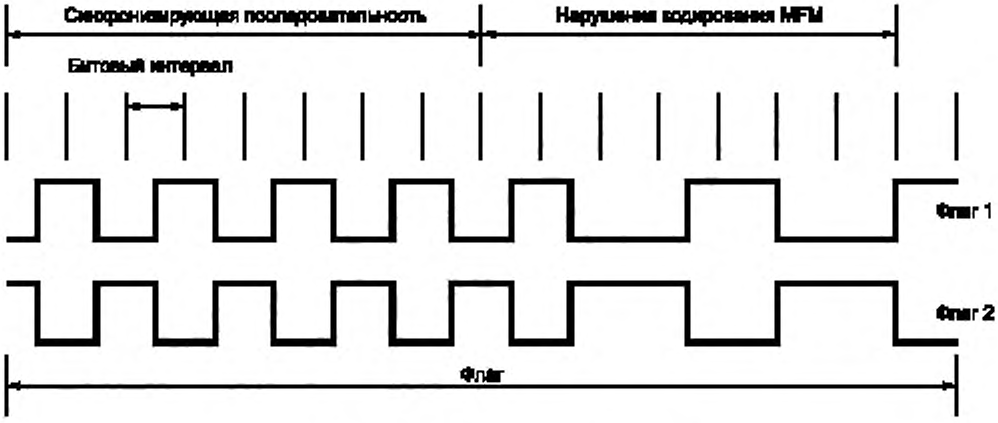



Рисунок 2 — Два возможных флага ответа

**6.2.3 Описание метода работы**

6.2.3.1 Общие положения

Данный подраздел определяет характеристики радиоинтерфейса между УСО и радиочастотной меткой. В нем подробно описаны передача энергии и двунаправленная связь между УСО и радиочастотной меткой.

Радиочастотные метки могут быть пассивными. В этом случае, энергия передается от УСО к радиочастотной метке высокочастотным магнитным полем с помощью антенн, установленных как в УСО, так и в радиочастотной метке. Частота силового питающего поля, fc, составляет 13,56 МГц ± 7 кГц. УСО должен быть способен обеспечить электропитание радиочастотной метки при любом положении метки внутри создаваемого им поля.

Команды передаются от УСО к радиочастотной метке путем фазовой модуляции колебаний (PJM) силового питающего поля. Фазовая модуляция колебаний PJM передает данные в виде очень малых изменений фазы силового питающего поля. Эти фазовые изменения находятся в диапазоне между ±3,0° и ±4,0°. Модуляция PJM не приводит к снижению мощности передачи. Полоса пропускания при модуляции PJM не шире, чем оригинальный двусторонний спектр данных. Уровень сигналов и скорость передачи данных в битах в боковых спектрах независимы друг от друга. Это позволяет задать любой уровень сигнала в боковой полосе, не влияя на скорость передачи данных в битах. Модуляция флуктуации фазы описана в приложении В.

Скорость передачи команд, кодируемых с использованием модифицированной частотной модуляции (MFM), составляет 423,75 Кбит/с.

УСО может быть дуплексным или полудуплексным. Если УСО дуплексный, то он может одновременно передавать команды и принимать ответы от радиочастотных меток. Радиочастотные метки полудуплексные.

Радиочастотная метка отвечает УСО посредством индуктивной связи, при этом напряжение в катушке индуктивности антенны радиочастотной метки модулируется поднесущей. Поднесущая получается путем деления частоты силового питающего поля.

Радиочастотные метки могут выбрать одну из восьми поднесущих частот в диапазоне от 969 кГц до 3013 кГц. Скорость передачи данных метки в битах, закодированных с помощью модифицированной частотной модуляции (MFM) и модулирования поднесущей методом двоичной фазовой манипуляции (BPSK) — 105,9375 Кбит/с.

Чтобы гарантировать, что сигналы от меток, отвечающих на разных каналах, могут быть приняты одновременно, сигналы должны иметь ограниченный частотный диапазон, чтобы уменьшить уровни гармоник данных (несущего сигнала) и поднесущих.

6.2.3.2 Интерфейс канала связи «УСО-радиочастотная метка»

Команды передаются от УСО к радиочастотной метке модуляцией флуктуации фазы силового питающего поля. Скорость передачи команды составляет 423,75 Кбит/с, все команды закодированы с помощью модифицированной частотной модуляции MFM до модулятора PJM.

6.2.3.2.1 Модуляция

Команды передаются от УСО к радиочастотной метке модуляцией флуктуации фазы силового питающего поля. При модуляции PJM данные передаются в виде очень малых инверсий фазы силового питающего поля. Это позволяет задать любой уровень сигнала в боковой полосе, не влияя на скорость передачи данных в битах.

Радиочастотные метки работают согласно своему назначению с минимальными по уровню сигналами, с использованием PJM, в боковых полосах, в соответствии с правилами FCC и ETSI.

Сдвиг фазы сигнала магнитного поля УСО с использованием модуляции флуктуации фазы иллюстрируется на рисунке 3 и в таблице 5.

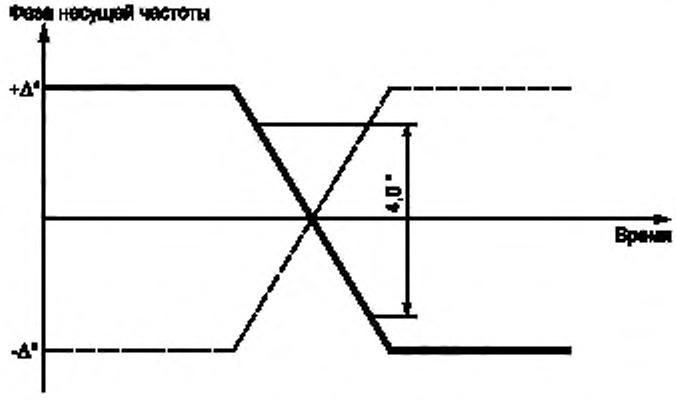



Рисунок 3 — Схема модуляции команды

Таблица 5 — Параметры модуляции команды

| Параметр | Минимум | Максимум | Единица измерения |
|---|---|---|---|
| Сдвиг фазы +Δ°, –Δ° | 3,0 | 4,0 | градус |
| Время, после которого сдвиг фаз должен остаться на уровне выше 95 % | 0,0 | 1,0 | мкс |

Примечание — Общий сдвиг фаз на 4,0° должен быть завершен в течение времени не более 500 нс. Сдвиг фаз не может превышать окончательное значение более чем на 5 %. Время изменения сдвига фаз между уровнями 10 % и 90 % (и наоборот) не должно превышать 40 нс. Величина сдвига фаз должна оставаться постоянной во время всей команды.

6.2.3.2.2 Скорость передачи данных в битах и кодирование данных

Скорость передачи кодированных команд составляет 423,75 Кбит/с (fs/32). Период битового интервала, используемый при кодировании команд, составляет 2,3599 мкс.

Все команды закодированы с помощью метода модифицированной частотной модуляции MFM до модулятора PJM. Биты кодируются с использованием правил кодирования модифицированной частотной модуляции MFM. MFM имеет самую узкую полосу пропускания из двоичных методов кодирования. Значение бита определяется по изменению состояния. Эти правила кодирования, определяются следующим образом:

- бит 1 определяется как изменение состояния в середине битового интервала;
- бит 0 определяется как изменение состояния в начале битового интервала,
- там, где бит 0 непосредственно следует за битом 1, изменения состояния нет.

Пример кодирования команды модифицированной частотной модуляцией и синхронизации двоичной последовательности 000100 показан на рисунке 4.

Как правило, фронты импульсов, показанных на рисунке 4, и на рисунке 5, должны быть синхронизированы с силовым питающим полем. Если не синхронизированы, то импульсы должны быть сгенерированы в УСО с точностью ±0,04 мкс относительно временных значений, показанных на рисунке.

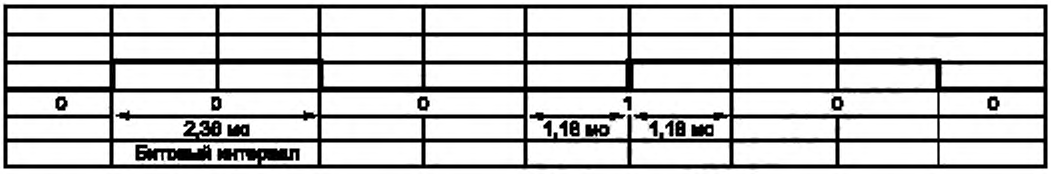



Рисунок 4 — MFM кодирование команды и временные диаграммы двоичной последовательности **000100**

6.2.3.2.3 Кадры данных в канале связи «УСО-радиочастотная метка»

Флаг команды определяет начало команды и длительность битового интервала. Флаг состоит из трех частей:

- синхронизирующая последовательность из 8-и бит действительных данных MFM;
- в обычных данных нет нарушения кодирования модифицированной частотной модуляции (MFM). Нарушение представляет собой последовательность из 5 изменений состояния, разделенных соответственно: интервалом в 1 бит, интервалом в 2 бита, интервалом в 1,5 бита, интервалом в 2 бита;
- завершающий 0 определяет конец флага и начало команды.

Синхронизирующая последовательность, нарушение кодирования и завершающий 0 для двух возможных флагов команды показаны на рисунке 5.

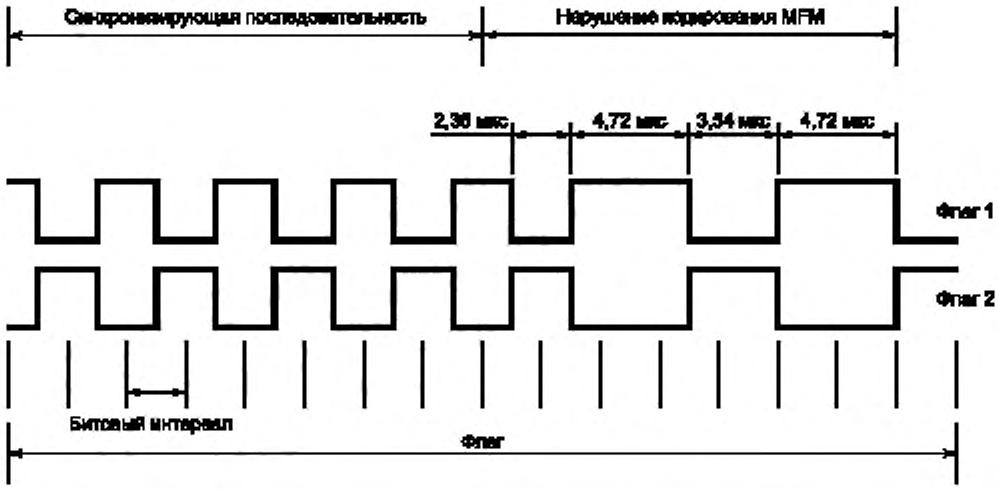



Рисунок 5 — MFM кодирование и временные диаграммы для двух возможных флагов команды

6.2.3.3 Интерфейс канала связи «радиочастотная метка-УСО»

Ответы закодированы методами MFM и BPSK. Радиочастотные метки используют для ответа одну из 8, на выбор, модулированных поднесущих частот.

6.2.3.3.1 Поднесущие

Радиочастотная метка может использовать одну из восьми поднесущих. Поднесущие получаются путем деления частоты силового питающего поля. Частоты канала и коэффициенты деления приведены в таблице 6.

Таблица 6 — Поднесущие частоты канала и коэффициенты деления

| Канал | Частота поднесущей, кГц | Коэффициент деления |
|---|---|---|
| A | 969 | 14,0 |
| B | 1233 | 11,0 |

*Окончание таблицы 6*

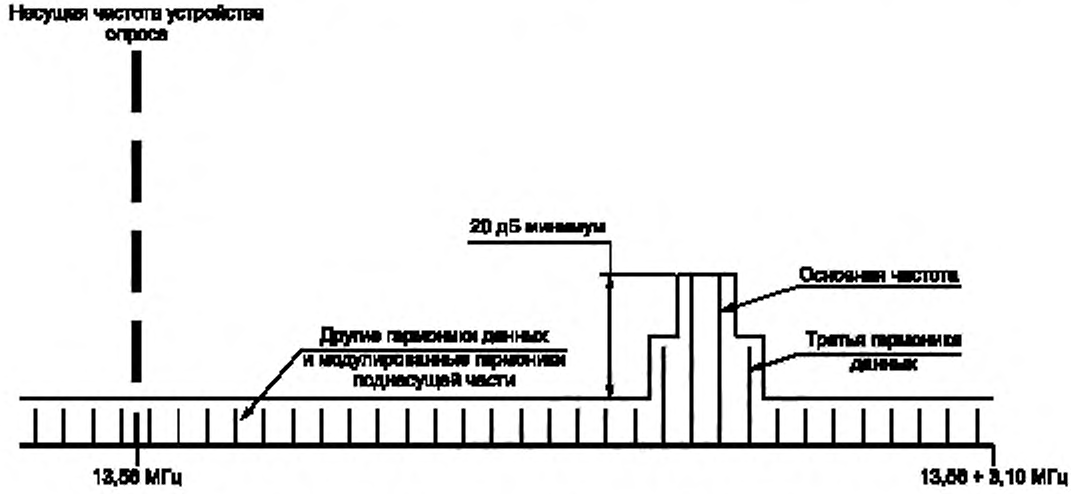
| Канал | Частота поднесущей, кГц | Коэффициент деления |
|---|---|---|
| C | 1507 | 9,0 |
| D | 1808 | 7,5 |
| E | 2086 | 6,5 |
| F | 2465 | 5,5 |
| G | 2712 | 5,0 |
| H | 3013 | 4,5 |

6.2.3.3.2 Модуляция

Радиочастотная метка отвечает УСО посредством индуктивной связи, при этом напряжение в катушке индуктивности антенны радиочастотной метки модулируется поднесущей. Модуляция основана на модуляции импеданса. Закодированные данные модулируются на поднесущей частоте как двоичная фазовая модуляция.

Системы радиочастотной идентификации должны соответствовать российскими правилами использования радиочастотного спектра. Чтобы гарантировать, что сигналы от меток, отвечающих на разных каналах, могут быть приняты одновременно, сигналы должны иметь ограниченный частотный диапазон, чтобы уменьшить уровни гармоник данных (несущего сигнала) и поднесущих.

Примечание — Например, спектр ответного сигнала радиочастотной метки для закодированного методом модифицированной частотной модуляции потока нулей, передаваемого со скоростью 105,9375 Кбит/с, может быть определен маской ответа радиочастотной метки, представленной на рисунке 6. Такое ограничение маски принято для регионов ITU 1, 2 и соответствует нормам Японии.




Примечание — Этот рисунок показывает верхнюю боковую полосу частот ответа радиочастотной метки. Такая же маска должна быть применена и к нижней боковой полосе частот.

Рисунок 6 — Маска ответного сигнала радиочастотной метки

6.2.3.3.3 Скорость передачи данных в битах и кодирование данных

Скорость передачи данных в битах закодированного ответного сигнала радиочастотной метки составляет 105,9375 Кбит/с ($f_c$/128). Длительность битового интервала, используемого для кодирования команд, составляет 9,4395 мкс. Используемая модуляция — BPSK (двоичная фазовая манипуляция).

Радиочастотная метка отвечает, используя одну из восьми выбранных модулированных поднесущих частот. Поднесущие частоты получены путем деления частоты силового питающего поля. Ответы кодируются с использованием MFM и модулируются на поднесущей методом двоичной фазовой манипуляцией (BPSK).

Правила кодирования MFM приведены в 6.2.3.2.2. Пример ответного сигнала, закодированного с помощью MFM, представляющего собой двоичную последовательность 000100, показан на рисунке 7. Как правило, фронты импульсов, показанных на рисунке 7 и на рисунке 8, должны быть синхронизированы с силовым питающим полем. Если не синхронизированы, то импульсы должны быть сгенерированы в радиочастотной метке с точностью ± 0,15 мкс относительно временных значений, показанных на рисунках.

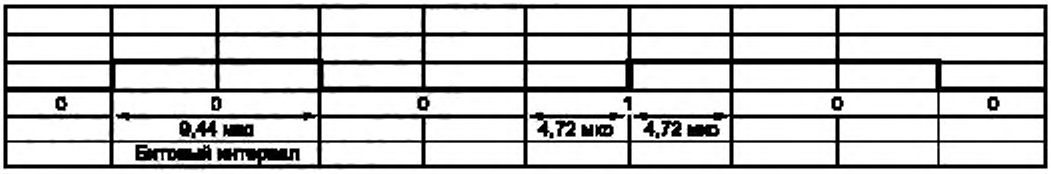



Рисунок 7 — MFM кодирование ответа и временные диаграммы двоичной последовательности 000100

6.2.3.3.4 Кадры данных в канале связи «радиочастотная метка-УСО»

Флаг ответа определяет начало ответа и длительность битовых интервалов. Флаг ответа состоит из трех частей:

- синхронизирующая последовательность из 9 бит действительных данных MFM;
- в обычных данных нет нарушения кодирования модифицированной частотной модуляции (MFM). Нарушение представляет собой последовательность из 4 изменений состояния, разделенных соответственно: интервалом в 2 бита, интервалом в 1,5 бита, интервалом в 2 бита. Четвертый (последний) переход определяет начало битового интервала;
- завершающий 0 в конце строки данных.

Синхронизирующая последовательность, нарушение кодирования и завершающий ноль для двух возможных флагов ответа показаны на рисунке 8.

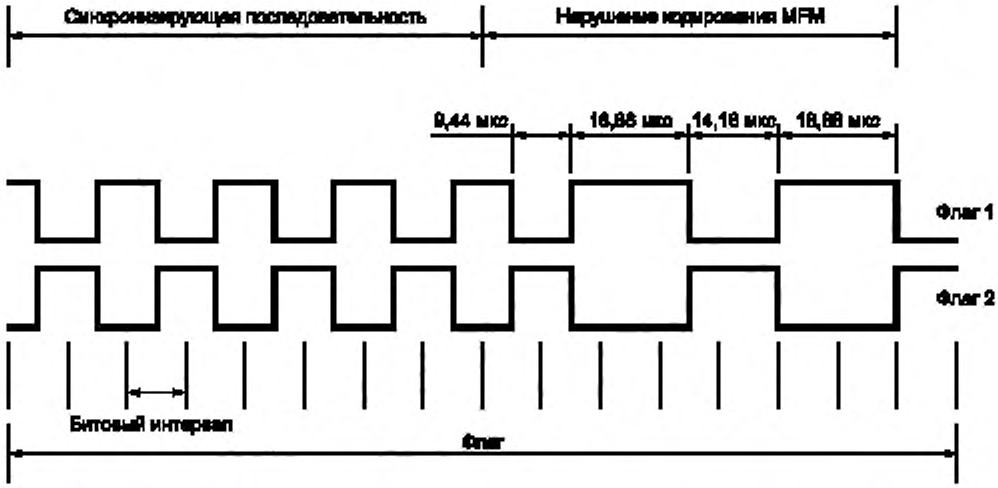



Рисунок 8 — MFM кодирование и временные диаграммы для двух возможных флагов ответа

### 6.2.4 Параметры протокола

Параметры протокола приведены в таблице 7.

Таблица 7 — Параметры протокола

| Позиция | Параметр | Описание |
|---|---|---|
| M2-P: 1 | «Кто передает первым» | Система ITF («УСО передает первым»). Радиочастотная метка не может ответить, пока не получит действительную команду от УСО |
| M2-P: 2 | Возможность адресации к радиочастотной метке | Есть. К радиочастотным меткам можно адресоваться как индивидуально, так и к группе |
| M2-P: 3 | Идентификатор радиочастотной метки | — |
| M2-P: 3a | Длина идентификатора радиочастотной метки | 64 бита (32-битовый специальный идентификатор, 16-битовый идентификатор группы применения и 16-битовый идентификатор изготовителя) |
| M2-P: 3b | Формат идентификатора радиочастотной метки | См. 6.2.5.6 к 6.2.5.9 |
| M2-P: 4 | Размер считывания | От 2 байт (16 бит) до максимального размера памяти |
| M2-P: 5 | Размер записи | Минимальный и максимальный размер записи зависят от типа памяти, в соответствии с инструкцией, определенной в 6.2.5.14 |
| M2-P: 6 | Время операции считывания | 1,282 мс + 150 мкс на 16 бит (2 байта) |
| M2-P: 7 | Время операции записи | 1,282 мс + 75,5 мкс на 16 бит (2 байта), без учета времени стирания памяти и времени записи |
| M2-P: 8 | Обнаружение ошибок | 16-битовый CRC (циклический избыточный код) в канале связи «УСО-радиочастотная метка», 32-битовый CRC в канале связи «радиочастотная метка-УСО» |
| M2-P: 9 | Исправление ошибок | Нет |
| M2-P: 10 | Объем памяти | Зависит от продукта. Никаких технических ограничений |
| M2-P: 11 | Структура команд и расширяемость | Поле команды составляет 16 бит, расширяемое без ограничений блоками по 16 бит. В настоящее время имеются 8 типов команд, каждая с 16 типами расширения |

### 6.2.5 Описание метода работы протокола

6.2.5.1 Общая информация

Радиочастотные метки могут быть пассивными, в этом случае энергия передается от УСО к радиочастотной метке высокочастотным (HF) магнитным полем с помощью индуктивно связанных антенн УСО и радиочастотной метки.

Диалог между УСО и радиочастотной меткой осуществляется на основе протокола ITF. После активации радиочастотной метки силовым питающим полем радиочастотная метка ожидает действующие команды без передачи данных. После получения команды радиочастотная метка передает ответ на команду.

Используется фазовая модуляция колебаний (PJM) (описана в 6.2.3 выше).

Память радиочастотной метки может быть расширена и составлять более 1 Мбит, таким образом система является расширяемой, если такие требования к радиочастотной метке предъявляются.

Идентификация нескольких меток выполняется за счет использования комбинации Множественного доступа с частотным разделением и Множественного доступа с временным разделением (FTDMA). Имеются восемь каналов ответа, доступных для использования радиочастотными метками. В ответ на действительную команду каждая радиочастотная метка случайным образом выбирает канал, по кото-

рому может передать свой ответ. Ответ передается по выбранному каналу один раз. После получения следующей команды каждая радиочастотная метка снова случайным образом выбирает канал и передает ответ. Этот метод псевдослучайной перестройки частоты ответа используется для каждой последующей действительной команды. В дополнение к случайному выбору канала радиочастотная метка может также случайным образом блокировать свой ответ. Когда ответ заблокирован, радиочастотная метка не передает этот ответ. Случайная блокировка ответа необходима при идентификации большого количества меток. Все частотные и временные параметры FTDMA определяются командой.

Все команды имеют временну́ю метку, радиочастотные метки хранят самую первую отметку, которую получили от УСО. Сохраненная временна́я метка определяет точно, когда радиочастотная метка впервые оказалась в поле УСО, и это представляет собой высокоточный метод определения порядка появления радиочастотных меток в поле УСО, не зависящий от скорости идентификации.

Временные настройки радиочастотной метки должны быть сохранены в памяти радиочастотной метки (например, используя память для временного хранения данных с произвольным доступом TRAM), и сохраняют данные во время перебоев питания, вызванных переключением силового питающего поля при неблагоприятной ориентации меток в поле УСО.

6.2.5.2 Описание элементов данных

Считанные данные — данные, считанные из встроенной памяти радиочастотной метки в ответ на действительную команду.

Записанные данные — данные, записанные в память радиочастотной метки в ответ на действительную команду.

Сохраненные данные представляют собой данные, сохраненные в памяти радиочастотной метки.

Жестко закодированные неизменяемые данные — данные в виртуальном ПЗУ.

6.2.5.3 Структура памяти радиочастотной метки

Настоящий стандарт описывает память радиочастотной метки исключительно в виртуальных терминах и не имеет целью ограничить возможные физические реализации памяти радиочастотной метки.

Память радиочастотной метки разделена на три области, приведенные в таблице 8.

Таблица 8 — Области памяти радиочастотной метки

| Область памяти | Примечание |
|---|---|
| Область заводской системной памяти | Содержит все поля, которые записываются и блокируются при производстве и производственных испытаниях чипов/меток |
| Область Пользовательской системной памяти | Записывается и блокируется в соответствии с требованиями пользователя |
| Область Пользовательской памяти | Записывается и блокируется в соответствии с требованиями пользователя |

Радиочастотные метки с виртуальной памятью 4 Кбит или менее используют 8-битовую адресацию и поля длиной в 8 бит. Радиочастотные метки с виртуальной памятью более чем 4 Кбит используют как 8-, так и 16-битовую адресацию и длину полей.

Память радиочастотной метки включает в себя идентификаторы радиочастотных меток, настройки и пользовательские поля. Виртуальная карта памяти, см. таблицу 9, включает в себя определенные поля. Порядок следования битов любого поля таков, что младший значащий бит хранится в младшем значащем бите адреса виртуальной памяти.

Виртуальная память структурно организована и адресуется к 16-битовым словам. МОДА 2 предусматривает различные типы меток с различными размерами блока памяти, где блок — это одно 16-битовое слово или более. Команды считывания считывают ноль или более слов. Команды записи записывают целые слова, где количество слов, которые могут быть записаны, определяется организацией памяти.

Память может быть заблокирована. После блокировки память не может быть перезаписана.

6.2.5.4 Виртуальная память радиочастотных меток

Таблица 9 показывает структуру виртуальной памяти радиочастотной метки.

Таблица 9 — Карта виртуальной памяти

| Слово | Тип памяти | Замечания | Регистр | Битовое число | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | Заводская системная память | Определенные поля | RFM | Зарезервировано для изготовителя | | | | | | | | | | | | | | | |
| 1 | | | MC | Код изготовителя | | | | | | | | | | | | | | | |
| 2 | | | SID0 | Специальный идентификатор 0 | | | | | | | | | | | | | | | |
| 3 | | | SID1 | Специальный идентификатор 1 | | | | | | | | | | | | | | | |
| 4 | Системная память пользователя | Определенные поля | GID | Идентификатор группы приложений | | | | | | | | | | | | | | | |
| 5 | | | CID | Условный идентификатор | | | | | | | | | | | | | | | |
| 6 | | | CW | Конфигурационное слово | | | | | | | | | | | | | | | |
| 7 | Пользовательская память | Неопределенные поля, если пароль не требуется | PW0 | Пароль 0 | | | | | | | | | | | | | | | |
| 8 | | | PW1 | Пароль 1 | | | | | | | | | | | | | | | |
| 9 | | | PW2 | Пароль 2 | | | | | | | | | | | | | | | |
| 10 и выше | — | Неопределенные поля | — | — | | | | | | | | | | | | | | | |

Слова начиная с 10-го должны быть Пользовательской памятью, за которой следует оставшаяся системная память. Специальный идентификатор 0 и пароль 0 являются наименее значимыми словами из этих нескольких полей слов.

6.2.5.5 Указатель блокировки

Указатель блокировки — 16-битовое поле виртуальной системной памяти, используемое, чтобы предотвратить перезапись памяти радиочастотной метки. Поле указывает на слово в памяти. Все полные блоки памяти с адресами, меньшими, чем число, сохраненное в указателе блокировки, не могут быть перезаписаны. Команды УСО не могут уменьшить значение указателя блокировки.

6.2.5.6 Уникальная идентификация (ID радиочастотной метки)

Уникальная идентификация для МОДА 2 может быть реализована непрерывно во время производства. Уникальный идентификатор должен представлять собой логический 64-битовый блок, соответствующий *ГОСТ Р ИСО/МЭК 15963*.

Физическая реализация идентификатора на метке остается за изготовителем.

64 логических бита структурированы в три части. См. рисунок 9.

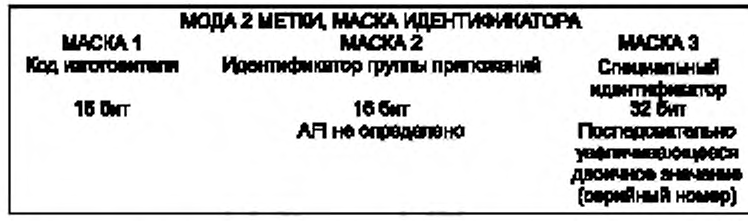



Рисунок 9 — Логическая организация идентификатора радиочастотной метки

6.2.5.7 Код изготовителя

Код изготовителя представляет собой маску из 16 битов.

Для обеспечения работоспособности системы требуется только, чтобы эта маска была заполнена двоичным значением (кодом).

Однако, чтобы однозначно идентифицировать изготовителей, код должен соответствовать *ГОСТ Р ИСО/МЭК 15963*.

Передача данных от радиочастотной метки в МОДА 2 должна производиться, начиная с младшего значащего бита, а данные должны располагаться, как показано в таблице 10.

Пример реализации согласуется с *ГОСТ Р ИСО/МЭК 15963*.

Код изготовителя — 16-битовое поле, устанавливается при производстве метки, на этапе тестирования. Структура кода изготовителя показана в таблице 10.

Таблица 10 — Поле кода изготовителя

MSB LSB

| 16 9 | 8 1 |
|---|---|
| 'E0' | Код изготовителя в соответствии с *ГОСТ Р ИСО/МЭК 7816-6* |

6.2.5.8 Идентификатор группы приложений (AFI)

Вторая маска состоит из 16 бит.

Для обеспечения работоспособности системы требуется только, чтобы эта маска была заполнена двоичным кодом.

Назначение данной маски — идентификация применений или множеств радиочастотных меток.

Рекомендуется, чтобы первый октет этой маски использовался для кодирования AFI, как определено в *[5]*, таким образом значение AFI должно быть определено в поле RA команды, как определено в *[6]*.

Второй октет этой маски не определен.

6.2.5.9 Специальный идентификатор

Третья маска — состоит из 32 бит.

Маска должна быть заполнена при производстве меток и должна содержать уникальное последовательно изменяющееся двоичное значение, определяемое изготовителем, и которое не может быть использовано им повторно. Это должно обеспечить уникальный идентификатор радиочастотной метки у данного изготовителя.

Комбинация специального идентификатора, Идентификатора семейства приложений и Кода изготовителя, должна обеспечить непрерывную и однозначную идентификацию радиочастотных меток.

Такой идентификатор позволяет однозначно идентифицировать радиочастотную метку и производить обмен данными именно с ней. Технологическое программное обеспечение, используемое при производстве меток, должно гарантировать, что значение уникального идентификатора увеличивается каждый раз, когда производится новая радиочастотная метка.

6.2.5.10 Условный идентификатор

В дополнение к идентификатору радиочастотной метки, память радиочастотной метки должна содержать дополнительную маску, известную как условный идентификатор (CID). Маска должна содержать 2 октета (16 бит).

Для обеспечения работоспособности системы требуется только, чтобы эта маска была заполнена двоичным кодом.

Условный идентификатор CID может быть присвоен при изготовлении (может представлять собой, например, дату изготовления) или может быть присвоен позже.

Настоящий стандарт не носит директивного характера относительно того, программировать условный идентификатор CID при изготовлении или позже, но изготовители должны решить, какой из этих вариантов будет реализован, и в соответствии с этим использовать условную идентификацию.

***Пример — Условный идентификатор CID может представлять собой код даты, который может быть использован как условие разрешения/запрещения доступа. Таким образом, например, радиочастотные метки, произведенные до или после определенной даты, могут либо использоваться, либо нет.***

6.2.5.11 Слово конфигурации

Слово конфигурации — 16-битовое поле, задаваемое пользователем. Конфигурация радиочастотной метки обычно устанавливается в соответствии с конкретным применением. Структура Слова конфигурации показана в таблице 11.

Таблица 11 — Поле Слова конфигурации

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| От 0 до 6 | RFU (зарезервировано) | — | Должны быть установлены в '0' |
| От 7 до 14 | Пользовательские настройки | — | — |
| 15 | Пароль | 0 | Пароль не требуется |
| 15 | Пароль | 1 | Требуется пароль |

6.2.5.12 Пароль

Поле пароля — 48-битовый идентификатор, задаваемый пользователем. Он используется, чтобы обеспечить уровень безопасности для доступа к памяти. Если радиочастотная метка сконфигурирована на режим «Пароль не требуется», область памяти пароля свободна и может быть использована как Пользовательская память. Если радиочастотная метка сконфигурирована на режим «Требуется пароль», пароль не может быть прочитан.

6.2.5.13 Пользовательская память

Распределение и заполнение Пользовательской памяти определяет пользователь.

6.2.5.14 Фиксированный код

Фиксированный код представляет собой 16-разрядные слова и включается в некоторые типы радиочастотных меток. Настоящий стандарт описывает фиксированный код исключительно в виртуальных терминах и не имеет целью ограничить возможные физические реализации фиксированного кода. Содержимое фиксированного кода определяется параметрами радиочастотных меток, в том числе общим объемом памяти и минимальным размером адресуемой памяти.

В нормальном ответе все слова, представляющие фиксированный код, передаются первыми, затем метка времени, а затем остальная часть ответа. Старший значащий бит всех слов фиксированного кода должен быть установлен в '1'. Структура фиксированного кода показана в таблице 12.

Таблица 12 — Поле фиксированного кода

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| От 0 до 6 | Параметр/ функция | $00_h$ | Размер памяти в 4 единицы слова (младший значащий бит) |
| | | $01_h$ | Размер памяти в 4 единицы слова (старший значащий бит) |
| | | $02_h$ | Размер блока памяти в словах |
| | | $03_h$ | Размер субблока памяти в словах |
| | | $04_h$ | Время стирания памяти + время записи в единицах по 100 мкс |
| | | От $05_h$ до $07_h$ | Зарезервированы для пользовательских фиксированных кодов |
| От 7 до 14 | Код/значение | От $00_h$ до $FF_h$ | Шестнадцатеричное значение или код, связанный с параметром или функцией |
| 15 | MSB (старший значащий бит) | | Должен быть установлен в '1' |

В рамках настоящего стандарта под фразой «время стирания + записи памяти» понимается время между окончанием последнего бита пришедшей команды записи и началом первого бита ответа радиочастотной метки. Также «размер памяти» должен определяться от 0 слова до верхней границы Пользовательской памяти.

Например, радиочастотная метка с характеристиками:

- 8192 бита памяти (512 слов, 128 * 4 единицы слова);
- размер блока памяти в 4 слова;
- минимальная область адресуемой памяти в 1 слово;
- время стирания памяти + время записи в 4 мс (40 * 100 мкс);

имела бы следующий фиксированный код:

- размер памяти $C000_h$;
- размер блока $8202_h$;
- минимальная область адресуемой памяти $8083_h$;
- время стирания + время записи $9404_h$.

В вышеупомянутом примере.

- старший значащий бит размера памяти не используется, поскольку все значения битов '0';
- минимальная область адресуемой памяти в 1 слово и размер блока памяти в 4 слова указывают, что разрешена запись в слова 1, 2, 3, и 4 памяти, и все записи должны быть в пределах указанной области (не должно быть никаких записей, выходящих за границы этой области);
- блок начинается со слова 0.

Память может быть заблокирована в рамках минимальной адресуемой области. Если интервал минимальной адресуемой памяти не задан, тогда память может быть заблокирована только поблочно.

6.2.5.14.1 Параметры радиочастотной метки «по умолчанию»

Если в фиксированном коде емкость памяти не определена, тогда емкость памяти установлена в 64 слова.

Если в фиксированном коде размер блока памяти не определен, тогда размер блока памяти установлен в 4 слова.

Если в фиксированном коде размер минимальной адресуемой памяти не определен, тогда размер минимальной адресуемой памяти установлен в 1 слово.

Если в фиксированном коде «время стирания + время записи» не определено, тогда «время стирания + время записи» составляет 5,0 мс или меньше.

6.2.5.15 Состояние блока безопасности

См. 6.2.5.5 и 6.2.5.12.

6.2.5.16 Описание методологии работы

Связь радиочастотной метки и УСО основывается на принципе ITF («УСО передает первым»). Радиочастотная метка может только ответить на команду, если полученная команда является действительной. Команды обычно используются для идентификации радиочастотных меток, считывания/записи данных или блокировки памяти. Команды также определяют тип ответа (короткий или нормальный) и режим ответа (фиксированный канал или случайный канал и т. д.). Короткий ответ используется, чтобы ускорить связь, тогда как нормальный ответ включает весь фиксированный код и некоторые области системной памяти.

Все радиочастотные метки, независимо от их типов, могут быть идентифицированы с помощью универсальной групповой команды. Универсальная групповая команда может настроить радиочастотные метки на передачу нормального ответа. Фиксированный код и данные системной памяти, включенные в нормальный ответ, предоставляют достаточно информации для того, чтобы пользователь мог продолжить опрос меток и передать другие допустимые команды.

Различные режимы ответа выбираются в соответствии с типами УСО (одно- или многоканальные) и ускоряют передачу в зависимости от количества радиочастотных меток в пределах рабочей области УСО.

6.2.5.16.1 Методы

См. 6.2.5.1 и 6.2.5.2.

6.2.5.16.2 Формат команды

Формат полей команды показан в полях таблицы 13. Все поля передаются, начиная с младшего значащего бита. Для многословных полей младший бит младшего значащего слова определяет младший значащий бит всего поля. Все команды, определенные в этом подразделе, являются «Обязательными командами», согласно классификации, приведенной в разделе 6.0.

Таблица 13 — Поля команды

| Код | Поле | Бит | Комментарий |
|---|---|---|---|
| F | Флаг | 16 | Нарушение последовательности MFM |
| Cd | Команда | 16 | Поле команды |
| Cn | Номер команды | 16 | Поле номера команды |

*Окончание таблицы 13*

| Код | Поле | Бит | Комментарий |
|---|---|---|---|
| SS | Специальный идентификатор | 32 | Поле идентификатора |
| G | Идентификатор группы приложений | 16 | Поле идентификатора |
| Ci | Условный идентификатор | 16 | Поле идентификатора |
| PPP | Пароль | 48 | Поле идентификатора |
| R | Считывать адрес и длину | 16 | 8-битовое поле адреса и 8-битовое поле длины для считывания из памяти |
| W | Записать адрес и длину | 16 | 8-битовое поле адреса и 8-битовое поле длины для записи в память |
| Ra | Считывать адрес | 16 | 16-битовое поле адреса для считывания из памяти |
| Rl | Считывать длину | 16 | 16-битовое поле длины для считывания из памяти |
| Wa | Записать адрес | 16 | 16-битовое поле адреса для записи в память |
| Wl | Записать длину | 16 | 16-битовое поле длины для записи в память |
| D | Записать данные | 16 | Данные для записи |
| C | CRC | 16 | Проверочный код CRC |

Допустимый формат команд показан в таблице 14, где:

- Поле пароля должно быть заполнено, только если этого требует метка;
- Показанные команды считывания-записи включают запись одного слова.

Таблица 14 — Допустимый формат команды

| Тип команды | Стартовые поля | Поля идентификатора | Поля адреса и длины | Данные | CRC |
|---|---|---|---|---|---|
| Групповое чтение | F [Cd] Cn | G Ci | [R] или [Ra Rl] | — | C |
| Специальное чтение | F [Cd] Cn | SS | [R] или [Ra Rl] | — | C |
| Групповая операция считывания/записи | F [Cd]Cn | G Ci \|PPP\| | [R W] или [Ra Rl Wa Wl] | D | C |
| Специальная операция считывания/записи | F [Cd]Cn | SS \|PPP\| | [R W] или [Ra Rl Wa Wl] | D | C |

Минимальная длина команды — 7 слов (112 бит).

При записи нулевой длины данные не передаются. При записи нескольких слов данных каждое записываемое слово сопровождается кодом CRC. Формат команды для записи двух слов показан в таблице 15.

Таблица 15 — Допустимый формат команды для считывания/записи нескольких слов

| Тип команды | Стартовые поля | Поля идентификатора | Поля адреса и длины | Данные | CRC | Данные | CRC |
|---|---|---|---|---|---|---|---|
| Групповая операция считывания/ записи | F [Cd] Cn | G Ci \|PPP\| | [R W] или [Ra Rl Wa Wl] | D | C | D | C |
| Специальная операция считывания/записи | F [Cd] Cn | SS \|PPP\| | [R W] или [Ra Rl Wa Wl] | D | C | D | C |

Для всех команд записи, если стартовое поле, поле идентификатора или поле длины и адреса переданы с ошибкой, данные не будут записаны в память радиочастотной метки.

Если хотя бы один из кодов CRC недействителен, то радиочастотная метка не будет отвечать.

6.2.5.16.3 Поля команды

6.2.5.16.3.1 Поле флага

Поле флага представляет собой нарушение кодирования MFM, которое отсутствует при нормальной передаче данных. Поле указывает на начало передачи команды.

6.2.5.16.3.2 Поле команды

Кодирование поля команды показано в таблице 16.

Таблица 16 — Побитовое кодирование поля команды

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| 0 | Тип команды | 0 | Команда считывания |
| | | 1 | Команда считывания/записи |
| 1 | Тип идентификатора | 0 | Целевая команда |
| | | 1 | Условная команда группы приложений |
| 2 | Тип ответа | 0 | Короткий ответ |
| | | 1 | Нормальный ответ |
| 3 | Фиксированный/случайный | 0 | Фиксированный канал ответа |
| | | 1 | Случайный канал ответа |
| От 4 до 6 | Выбор канала/коэффициент блокировок (канала связи) | 000 | [фиксированный канал A] или [случайный канал, канал связи свободен] |
| | | 001 | [фиксированный канал B] или [случайный канал с коэффициентом блокировок ½] |
| | | 010 | [фиксированный канал C] или [случайный канал, коэффициент блокировок ¾] |
| | | 011 | [фиксированный канал D] или [случайный канал, коэффициент блокировок 7/8] |
| | | 100 | [фиксированный канал E] или [случайный канал, коэффициент блокировок 31/32] |
| | | 110 | [фиксированный канал G] или [случайный канал, коэффициент блокировок 511/512] |
| | | 111 | [фиксированный канал H] или [случайный канал, полностью загружен] |
| 7 | Адрес и длина | 0 | 8-битовое поле адреса и 8-битовое поле длины |
| | | 1 | 16-битовое поле адреса и 16-битовое поле длины |
| 8 | Разрешение тестовой команды | 0 | Нормальная команда |
| | | 1 | Тестовая команда изготовителя* |
| От 9 до 13 | Зарезервировано RFU | — | Должны быть установлены в '0' для обязательных команд |
| 14 | Класс команды | 0 | Обязательная команда |
| | | 1 | Команда пользователя |

* *По классификации команд, приведенной в 6.0.*

*Окончание таблицы 16*

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| 15 | Расширение команды | 0 | Указывает радиочастотной метке, что это слово является последним полем команды |
| | | 1 | Указывает радиочастотной метке, что следующие слово является расширением поля команды |

Для того чтобы команда являлась допустимой, поля команды должны быть установлены согласно одной из комбинаций, показанных в таблице 16. Кроме того, для радиочастотных меток с виртуальной памятью размером 4 Кбита или меньше, поле «адреса и длины» (7 бит) должно быть установлено в '0'. Функция, разрешаемая 8-м битом, используется только во время тестирования пластины с микросхемами (при изготовлении), и должна быть заблокирована изготовителем, чтобы пользователю были доступны только стандартные команды.

Ниже описаны поля команды.

6.2.5.16.3.3 Тип команды

Поле «Тип команды» определяет, является ли команда командой считывания или командой считывания/записи.

Команда считывания используется для считывания данных из памяти радиочастотной метки. Для быстрой идентификации радиочастотной метки поле длины в команде считывания может быть установлено в ноль.

Команда считывания/записи используется для считывания данных из памяти радиочастотной метки и записи данных в память радиочастотной метки. Для выполнения только операции записи длина области считывания* устанавливается в ноль. Чтобы заблокировать память радиочастотной метки, длина области записи устанавливается в ноль, а адрес записи устанавливается равным наименьшему незаблокированному адресу памяти.

6.2.5.16.3.4 Тип идентификатора

Поле «Тип идентификатора» указывает, является ли команда определенной или условной командой группового применения.

Целевые команды используются для идентификации и обмена данными с отдельными радиочастотными метками.

Команды с групповыми параметрами применения (групповые) используются для идентификации и обмена данными с группой радиочастотных меток, удовлетворяющих заданному условию, или всеми группами меток, которые удовлетворяют заданному условию.

6.2.5.16.3.5 Тип ответа

Поле «Тип ответа» определяет, является ли ответ коротким или нормальным.

Короткий ответ используется, чтобы минимизировать время обмена данными.

Нормальный ответ используется, когда УСО требует, чтобы в ответ были включены фиксированный код и данные системной памяти.

6.2.5.16.3.6 Фиксированный/случайный

Поле «Фиксированный/случайный» определяет, как передается ответ радиочастотной метки: на фиксированном или случайном канале.

6.2.5.16.3.7 Выбор канала/коэффициент блокировок

Поле «Выбор канала/коэффициент блокировок» определяет, используется для передачи определенный фиксированный канал или случайный канал с указанным коэффициентом блокировок.

Это поле связано с полем «Фиксированный/случайный», описанным выше. Если выбран режим «Фиксированный канал», тогда поле «Выбор канала/коэффициент блокировок» указывает на конкретный канал, который должен использоваться в ответе. Если выбран режим «Случайный канал», тогда поле «Выбор канала/коэффициент блокировок» указывает на выбранный коэффициент блокировок ответов в канале связи**.

---

* *В поле команды «Адрес и длина».*

** *Или на режим «Свободный канал» — [000].*

Для допустимых команд в режиме «Случайный канал»:

- Если данные в поле команды соответствуют состоянию «Свободный канал» ([000]), то радиочастотная метка будет передавать ответы по случайно выбранным каналам.

- Если указан коэффициент блокировок, соответствующий значениям от $^{1}/_{2}$ до $^{511}/_{512}$, радиочастотная метка принимает решение передавать или не передавать данные по выбранному каналу самостоятельно и случайным образом. Коэффициент блокировок, содержащийся в команде ($^{1}/_{2}$, $^{3}/_{4}$ и т. д.), определяет вероятность принятия решения о блокировке ответа (отказе от передачи по выбранному каналу связи).

- Если радиочастотная метка полностью заблокирована, она не будет отвечать на команды и перейдет временно в состояние режима ожидания. Если радиочастотная метка находится в состоянии режима ожидания, она ответит только на команду УСО с новым идентификатором УСО*, см. 6.2.5.16.3.9.

6.2.5.16.3.8 Адрес и длина

Поле «Адрес и длина» определяет, содержит ли команда 8-битовое поле адреса и 8-битовое поле длины, или 16-битовое поле адреса и 16-битовое поле длины.

6.2.5.16.3.9 Номер команды

Поле «Номер команды» используется для того, чтобы установить отметку локального времени и идентифицировать УСО.

Кодирование поля «Номер команды» показано в таблице 17.

Таблица 17 — Поля «Номера команды»

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| От 0 до 7 | Отметки локального времени | — | Устанавливается по требованию УСО |
| От 8 до 14 | Идентификатор УСО | — | Устанавливается по требованию УСО |
| 15 | Старший значащий бит (MSB) | — | Должен быть установлен в '0' |

Старший значащий байт «номера команды» должен быть отличным от $00_h$.

Когда радиочастотная метка начинает обмен данными с новым УСО, она сохраняет значение «Номер команды», содержащийся в первой полученной действительной команде. Радиочастотная метка должна сохранить это значение в течение краткого перерыва, предоставляемого УСО. Все ответы радиочастотной метки содержат это сохраненное значение, которое называют меткой времени. Младший значащий байт значения «Номера команды» периодически увеличивается УСО и используется в последующих командах. Радиочастотная метка не обновляет временную метку. Поэтому временна́я метка показывает, когда радиочастотная метка впервые получила команду.

Радиочастотная метка определяет, что начинается обмен данными с новым УСО, когда старший значащий байт «Номера команды» (содержащийся во всех действительных командах) и временна́я метка изменились, или когда радиочастотная метка обнаруживает, что длительность прерывания излучения УСО превышает положенное время.

При одновременной работе нескольких радиочастотных меток, если две или более меток отвечают в одно и то же время на одном и том же канале, то ни один из ответов получен не будет. Если количество работающих меток достаточно велико, то возможно, что радиочастотной метке придется осуществить передачу несколько раз, прежде, чем будет выбран уникальный канал и ее ответ будет принят УСО. Существует также вероятность, что радиочастотная метка выберет уникальный канал уже при первой передаче. Таким образом, последовательность полученных от меток ответов, не может использоваться для определения последовательности, в которой они появлялись в поле УСО. Однако, метка времени, содержащаяся в ответе, дает правильную последовательность появления меток в поле УСО.

6.2.5.16.3.10 Идентификаторы радиочастотной метки

6.2.5.16.3.11 Поле специального идентификатора

Специальный идентификатор используется, чтобы обмениваться данными со специальными радиочастотными метками. Для того чтобы команда была действительной, специальный идентификатор в команде должен соответствовать специальному идентификатору, прописанному в радиочастотной метке.

---

* *Или по истечении определенного периода отсутствия поля УСО.*

6.2.5.16.3.12 Поле идентификатора группы приложений

Идентификатор группы приложений используется, чтобы обмениваться данными только с радиочастотными метками одной группы приложений, либо с радиочастотными метками всех групп приложений. Для того чтобы команда была действительной, идентификатор группы приложений, содержащийся в команде, должен быть равен $FFFF_h$, либо должен строго соответствовать идентификатору группы приложений, сохраненному в радиочастотных метках, с которыми устанавливается связь.

Если идентификатор команды равен $FFFF_h$, а остальная часть команды также является действительной, то возможен обмен данными с радиочастотными метками любой группы приложений.

6.2.5.16.3.13 Поле условного идентификатора

Условный идентификатор используется, чтобы связаться с радиочастотными метками, которые удовлетворяют некоторому условию. Для того чтобы команда была действительной, условный идентификатор, содержащийся в команде, должен быть меньше или равен условному идентификатору, сохраненному в радиочастотной метке.

6.2.5.16.3.14 Поле пароля

Пароль используется для того, чтобы ограничить возможность записи в память радиочастотной метки. Для того чтобы команда была действительной, пароль должен присутствовать в команде только в случае, если радиочастотная метка сконфигурирована в состояние «требуется пароль», а команда должна быть командой считывания/записи. Для того чтобы команда считывания/записи при обмене данными с защищенной паролем меткой была действительной, пароль в команде должен быть идентичен паролю, сохраненному в радиочастотной метке.

6.2.5.16.3.15 Поле адреса и длины

Поле команды определяет, содержит ли команда 8-битовый адрес и 8-битовую длину блока данных, или 16-битовый адрес и 16-битовую длину.

Поле адреса и длины определяет начальный адрес и длину блока данных «в словах» при чтении и записи. Для того чтобы команда была действительной, поле адреса и длины должны указывать на данные в памяти только в допустимом диапазоне адресов памяти. Допустимыми адресами памяти являются следующие:

- Для считывания информации из меток, сконфигурированных как «Пароль не требуется»: адреса, соответствующие диапазону от «0» до наибольшего адреса Пользовательской памяти.

- Для считывания информации из меток, сконфигурированных как «Требуется пароль»: адреса, соответствующие диапазону от «10» до наибольшего адреса Пользовательской памяти.

- Для записи в память: от адреса, сохраненного в Указателе блокировки, до наибольшего адреса Пользовательской памяти.

- При блокировке памяти радиочастотной метки: от адреса, сохраненного в Указателе блокировки, и выше.

- Чтобы заблокировать память радиочастотной метки, длина записи устанавливается в «0», а адрес записи устанавливается на самый младший незаблокированный адрес памяти, если команда является действительной, то радиочастотная метка установит Указатель блокировки равным адресу записи.

- Кроме того, для того чтобы команда была действительной, поля адреса и длины должны быть установлены так, как этого требуют различные типы меток, чтобы соответствовать их ограничениям при обращении к блокам памяти.

Если это однозначно определено в спецификации изготовителя микросхемы:

- допустимые адреса памяти при считывании памяти могут включать системную память, расположенную в области Пользовательской памяти,

- допустимые адреса памяти при считывании памяти могут включать слова от 0 до 6 для меток, сконфигурированных как «Требуется пароль».

6.2.5.16.3.16 8-битовое поле адреса и 8-битовое поле длины

В тех случаях, когда в поле команды указаны 8-битовое поле адреса и 8-битовое поле длины, они должны быть закодированы, как показано в таблице 18 и таблице 19.

Таблица 18 — Чтение из памяти с 8-битовым полем адреса и 8-битовым полем длины

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| От 0 до 7 | 8-битовое поле адреса | — | Установлено в соответствии с требованиями пользователя |
| От 7 до 15 | 8-битовое поле длины | — | Установлено в соответствии с требованиями пользователя |

Таблица 19 — Запись в память с 8-битовым полем адреса и 8-битовым полем длины

| Номер бита | Поле | Состояние | Описание |
|---|---|---|---|
| От 0 до 7 | 8-битовое поле адреса | — | Установлено в соответствии с требованиями пользователя |
| От 7 до 15 | 8-битовое поле длины | — | Установлено в соответствии с требованиями пользователя |

6.2.5.16.3.17 16-битовое поле адреса и 16-битовое поле длины

В тех случаях, когда в поле команды указаны 16-битовое поле адреса и 16-битовое поле длины, они должны иметь длину по 16 бит каждое.

6.2.5.17 Записываемые данные

Записываемые данные — данные, которые будут записаны в радиочастотную метку.

6.2.5.18 CRC (циклический избыточный код)

Все команды CRC рассчитываются от конца поля флага.

Используемый алгоритм вычисления CRC 16-битовой команды представляет собой алгоритм IBM «Синхронное управление передачей данных» (Synchronous Data-Link Control SDLC), полиномиальный стандартизованный Международным консультативным комитетом по телеграфии и телефонии CCITT для использования в протоколе коммутации пакетов X.25, реализован в большинстве меток с последовательным интерфейсом обмена данными и описан в *[7]*. CRC должен вычисляться согласно *[7]*.

Используется следующий алгоритм:

$$g(X) = X^{16} + X^{12} + X^5 + 1.$$

Алгоритм генерации немного расходится с базовым алгоритмом CRC16, в котором генератор предварительно загружен единичными битами — '1', а не нулевыми — '0'. В дополнение проверочные биты инвертируются перед передачей. Соответственно, действительное сообщение не распознается по остаткам нулей (как это имеет место с CRC16), а распознается по определенной константе.

Пример команды УСО и CRC:

**поле флага**

| | |
|---|---|
| поле команды | 0000 |
| номер команды | 1234 |
| специальный идентификатор 0 | 1234 |
| специальный идентификатор 1 | 5678 |
| адрес и длина при считывании | 1001 |
| CRC | 8C16 |

Метод вычисления и пример этого 16-битового CRC даны в приложении М.

6.2.5.19 Формат ответа

Поля ответа приведены в таблице 20. Все поля передаются, начиная с младшего значащего бита. Для многобайтовых полей младший значащий бит младшего значащего байта определяет младший значащий бит поля.

Таблица 20 — Поля ответа

| Код | Поле | Биты | Комментарий |
|---|---|---|---|
| F | Флаг | 16 | Нарушение кодирования в MFM последовательности |
| H | Фиксированный код | 16 | Поле фиксированного кода |
| T | Метка времени | 16 | Поле идентификатора |
| L | Указатель блокировки | 16 | Поле идентификатора |

*Окончание таблицы 20*

| Код | Поле | Биты | Комментарии |
|---|---|---|---|
| M | Код изготовителя | 16 | Поле идентификатора |
| SS | Специальный идентификатор | 32 | Поле идентификатора |
| G | Идентификатор группы приложений | 16 | Поле идентификатора |
| Ci | Условный идентификатор | 16 | Поле идентификатора |
| Co | Слово конфигурации | 16 | Поле идентификатора |
| D | Считываемые данные | 16 | Считываемые данные |
| CC | CRC | 32 | Проверка CRC |

Формат допустимых ответов приведен в таблице 21.

Таблица 21 — Допустимый формат ответа

| Тип ответа | Стартовые поля | Поля системной памяти | Данные | CRC |
|---|---|---|---|---|
| Нормальный | F [H] T | L M SS G Ci Co | [D] | CC |
| Короткий | F T | SS | [D] | CC |

Минимальная длина ответа составляет 96 бит. Тип ответа определяется командой УСО.

6.2.5.20 Формат ответа

6.2.5.20.1 Поле флага

Поле флага представляет собой нарушение кодирования MFM, которое отсутствует при нормальной передаче данных. Поле указывает на начало передачи команды.

6.2.5.20.2 Поле Фиксированного кода, поле идентификатора

Если конструкция радиочастотной метки предусматривает наличие Фиксированного кода*, то все данные Фиксированного кода передаются в нормальных ответах. Старший значащий бит фиксированного кода установлен в '1', а старший значащий бит метки времени установлен в '0'. Таким образом, УСО может обнаружить окончание данных фиксированного кода, проверяя старший значащий бит полученных слов.

6.2.5.20.3 Поле временной метки

Временна́я метка используется, чтобы безошибочно определить порядок появления радиочастотных меток. Временна́я метка соответствует «Номеру команды», содержащемуся в первой действительной команде, полученной после появления радиочастотной метки в поле УСО.

6.2.5.20.4 Поля системной памяти

Поля системной памяти в ответе соответствуют содержимому системной памяти, сохраненному в виртуальной памяти радиочастотной метки.

6.2.5.21 Считываемые данные

Считываемые данные — данные, запрашиваемые действительной командой.

6.2.5.22 Поле CRC

Код CRC ответа вычисляется в конце стартового поля флага.

Код CRC ответа представляет собой 32-битовый «Ethernet CRC». Его свойства аналогичны свойствам 16-битового кода «IBM CRC», используемого в команде маршрутизации, в которой маршрут загружается в регистр, на выходе которого слово результата вычислений инвертируется, и представляют собой некоторую константу, отличную от нуля.

Используемый алгоритм следующий:

$$g(X) = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^{8} + X^{7} + X^{5} + X^{4} + X^{2} + X^{1} + 1.$$

* *Определен в 6.2.5.14.*

Пример ответа радиочастотной метки и код CRC:

**поле флага**

| | |
|---|---|
| временна́я метка | 1234 |
| специальный идентификатор 0 | 1234 |
| специальный идентификатор 1 | 5678 |
| считываемые данные | ABCD |
| CRC | E8C5 8742 |

Описание метода и пример этого 32-битового кода CRC даны в приложении N.

6.2.5.23 Состояния радиочастотной метки

6.2.5.23.1 Диаграмма состояния

Радиочастотная метка может быть в одном из четырех состояний:

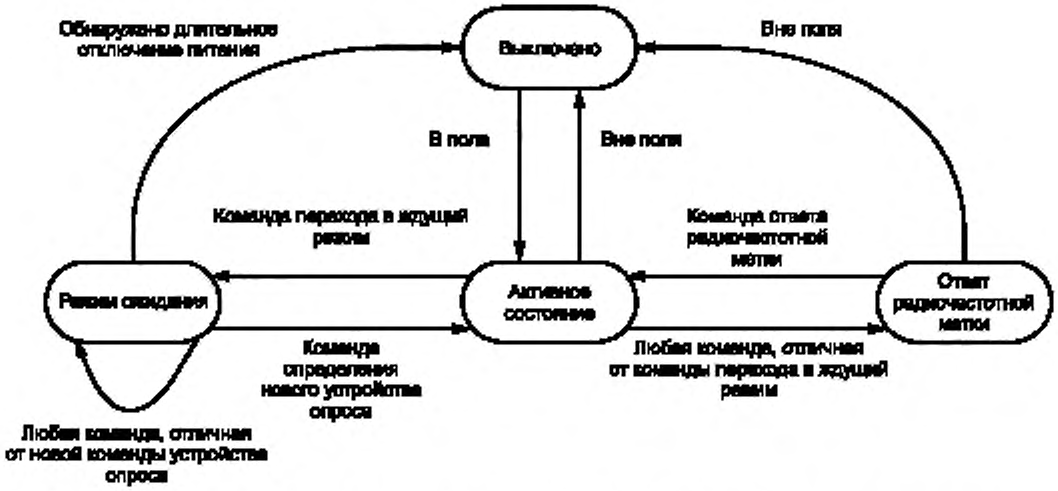
- выключено (питание отключено) (Power Off). Радиочастотная метка находится в состоянии «выключено», когда она не может быть активирована УСО. Выключение также достигается в результате обнаружения длительного отключения питания (LPB);

- активное состояние (Active). Радиочастотная метка находится в активном состоянии, когда она активирована УСО. В этом состоянии она должна обрабатывать любую команду УСО;

- ответ радиочастотной метки (Tag Reply). Радиочастотная метка находится в состоянии «ответ радиочастотной метки», когда она получает действительную команду. Если радиочастотная метка остается во включенном состоянии, то она будет находиться в этом состоянии до тех пор, пока не завершит ответ, после чего она вернется в активное состояние;

- режим ожидания (Fully Muted). Радиочастотная метка переходит в режим ожидания, когда она получает действительную команду, содержащую информацию о полной загруженности канала связи. Если радиочастотная метка остается во включенном состоянии, то она будет находиться в режиме ожидания до тех пор, пока не получит новую действительную команду от УСО.

Переход между этими состояниями показан на рисунке 10.



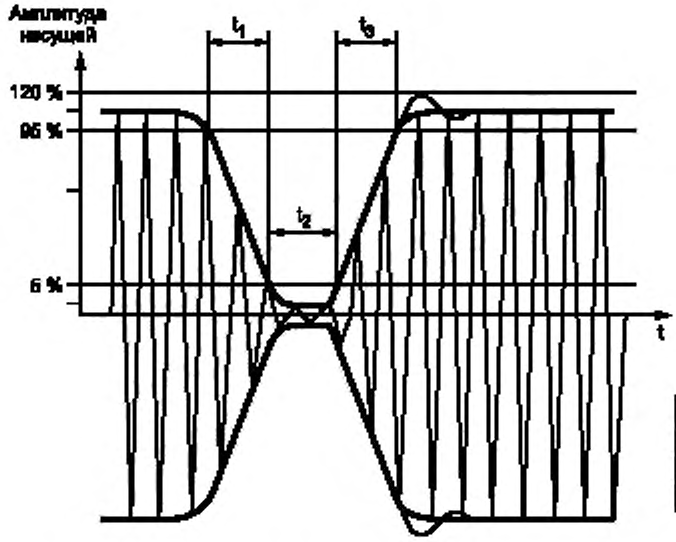

Примечание — Для недопустимых команд радиочастотная метка останется в своем текущем состоянии.

Рисунок 10 — Диаграмма изменений состояния радиочастотной метки

6.2.5.23.2 Кратковременное отключение питания

Перерыв в обнаружении радиочастотной меткой несущей от УСО в 5 мкс или более, но меньше, чем значение «Длительное отключение питания» (LPB), см. ниже, рассматривается как «Короткое отключение питания» (SPB). Если происходит «Короткое отключение питания», то радиочастотная метка инициализируется и затем переходит в активное состояние.

На рисунке 11 описывается прерывание сигнала несущей УСО.




| | Min, мкс | Max, мкс |
|---|---|---|
| $t_1$ | 0,0 | 10,0 |
| $t_2$ | 6,0 | |
| $t_3$ | 0,0 | 10,0 |

Рисунок 11 — Отключение силового питающего поля УСО

6.2.5.23.3 Длительное отключение питания

Радиочастотная метка имеет память для временного хранения данных с произвольным доступом TRAM или эквивалент, см. ниже. Если питание радиочастотной метки отключается на длительное время, данные в памяти TRAM могут быть повреждены. Используя детектор длительного отключения питания (LPB), радиочастотная метка может обнаруживать отключения питания достаточной длительности, которые способны привести к повреждению данных в TRAM. В результате таких отключений данные будут очищены.

Оперативная память TRAM или эквивалент должны оставаться действующими по крайней мере 50 мс. Поэтому Длительное отключение питания — это отключения питания на 50 мс или более.

6.2.5.23.4 TRAM (Память для временного хранения данных с произвольным доступом)

В настоящем стандарте приведены требования к временному хранению данных. Данные требования реализованы в технологии TRAM, упомянутой в разделах настоящего стандарта. Для этих целей могут использоваться также другие технологии. В настоящем разделе приведено описание технологии TRAM для тех пользователей, которые выбрали эту технологию. Память TRAM, о которой упоминается в этой МОДА, является энергозависимой памятью, такой как SRAM или DRAM, которые были разработаны, чтобы максимизировать время хранения информации после того, как будет отключено питание. Если питание подать прежде, чем TRAM разрядится, память будет автоматически обновлена.

TRAM обеспечивает мгновенную запись и кратковременное хранение информации. Малое время записи необходимо, чтобы установить режим ожидания и записать метку времени и другие временные настройки.

Примечание — При включении питания радиочастотная метка проверяет состояние детектора LPB, если выясняется, что произошло длительное отключение питания, радиочастотная метка очистит всю память TRAM.

6.2.5.23.5 Режим ожидания

Когда предметы, промаркированные радиочастотными метками, движутся по конвейеру, положение и ориентация меток неконтролируемы. Для того чтобы УСО на конвейере имел возможность запитать радиочастотные метки и связаться с ними, независимо от их положения и ориентации, он должен генерировать поле, которое бы циклически переключалось между осями *X*, *Y* и *Z*. Следствием воздействия циклического поля является периодическое отключение питания меток. Во время этих отключений электроэнергии любая информация, находящаяся в энергозависимой памяти радиочастотной метки, не должна быть потеряна.

Особое внимание должно уделяться управлению перебоями в подаче электроэнергии при эксплуатации нечувствительных к ориентации УСО. Например, в случае, когда идентифицируются несколько меток, существует требование, чтобы уже проидентифицированные радиочастотные метки были переведены в режим ожидания или были временно не активны, чтобы не препятствовать идентификации остальных меток.

Такие методы, как Память для временного хранения данных с произвольным доступом (TRAM), используются для обеспечения перевода меток во временно не активное состояние во время перебоев в подаче электроэнергии.

**6.2.6 Параметры системы управления коллизиями**

Параметры системы управления коллизиями приведены в таблице 22.

Таблица 22 — Таблица параметров системы управления коллизиями

| Позиция | Параметр | Описание |
|---|---|---|
| M2-A: 1 | Тип (вероятностный или детерминированный) | Вероятностный |
| M2-A: 2 | Линейность | Линейный для количества меток менее 10 000 меток<br>Полиномиальный для количества меток более 10 000 меток |
| M2-A: 3 | Емкость инвентаризации радиочастотных меток | Более чем 32 000 на каждый идентификатор группы приложений/условный идентификатор |
| M2-A: 4 | Скорость идентификации нескольких радиочастотных меток | 100 меток за 150 мс |

**6.2.7 Описание параметров системы управления коллизиями (справочное)**

6.2.7.1 Общее описание

В данной МОДА идентификация нескольких меток выполняется путем использования комбинации Множественного доступа с частотным разделением и Множественного доступа с временным разделением (FTDMA).

Существует восемь каналов ответа, доступных для использования радиочастотными метками. В ответ на действительную команду каждая радиочастотная метка случайным образом выбирает канал, по которому будет передавать ответ. Ответ передается один раз по выбранному каналу. После получения следующей действительной команды каждая радиочастотная метка случайным образом выбирает новый канал и передает ответ, используя новый выбранный канал. Этот метод перестройки частоты ответа, использующий случайно выбранный канал, повторяется для каждой последующей действительной команды.

В дополнение к случайному выбору канала некоторые ответы радиочастотной метки могут быть случайным образом подавлены. Когда ответ подавлен, радиочастотная метка не будет передавать этот ответ. Случайное подавление необходимо при идентификации большого количества радиочастотных меток. Как только радиочастотная метка идентифицировалась, она временно переводится командой в «режим ожидания», после чего будет отвечать только в случае, описанном в 6.2.5.16.3.7.

Все частотные и временные параметры FTDMA определяются командой. FTDMA обеспечивает лучшую производительность по сравнению с технологией TDMA множественного доступа с разделением по времени, работающей на единственной частоте, потому что множественные ответы могут быть одновременно получены по различным каналам.

В дополнение к множеству меток, которые могут находиться в поле УСО, там также может находиться большое количество старых меток или меток с истекшим сроком. В приложениях, использующих одноразовые метки, количество старых меток или меток с истекшим сроком, может быть намного больше, чем количество рабочих меток. Высокоскоростная система идентификации РЧИ должна быть в состоянии идентифицировать рабочие радиочастотные метки, исключая или игнорируя старые радиочастотные метки и метки с истекшим сроком.

В такой системе радиочастотные метки имеют условный идентификатор. Поле условного идентификатора каждой радиочастотной метки запрограммировано при изготовлении радиочастотной метки. Поле может быть запрограммировано меткой времени и даты, которое будет сравниваться с условным идентификатором, передаваемым в каждой команде. Радиочастотные метки ответят на команды

только в том случае, если проверка условного идентификатора завершится успешно. Таким образом, старые радиочастотные метки и метки с истекшим сроком исключаются из процесса идентификации.

Если УСО дуплексный, то он может передавать команды и одновременно принимать ответы от восьми меток, притом что каждая радиочастотная метка передает на одном из восьми уникальных каналов.

6.2.7.2 Каналы ответа

Система использует восемь каналов ответа в частотном диапазоне от 969 кГц до 3013 кГц. Чтобы максимизировать скорость идентификации радиочастотной метки при различных типах УСО и радиочастотных меток используются различные режимы ответа. Режим ответа, используемый радиочастотной меткой, задается командой УСО.

6.2.7.3 Режимы ответа

6.2.7.3.1 Режим ответа на фиксированном канале

Если УСО выбирает режим ответа на фиксированном канале, то радиочастотная метка передаст свой ответ один раз по каналу, выбранному УСО. Этот режим может использоваться для УСО, работающих на одном канале, в случае, когда все радиочастотные метки — одного типа.

Этот режим может также использоваться в многоканальных УСО, чтобы управлять ответом идентифицируемых радиочастотных меток с помощью команд, определяющих различные фиксированные каналы для ответа. Используя этот режим, могут быть получены данные одновременно от восьми меток.

6.2.7.3.2 Режим ответа на случайном канале

6.2.7.3.2.1 Свободный канал

Если УСО выберет режим ответа на свободном канале, то радиочастотные метки будут передавать полный ответ один раз на канале, выбранном ими случайным образом. Этот режим может использоваться многоканальными УСО для небольшого количества меток.

Этот режим может также использоваться для работы с небольшим количеством меток одноканальными УСО. В случае одноканального УСО при работе с большим количеством радиочастотных меток на зафиксированном канале будут происходить коллизии. Использование режима ответа на свободном канале при работе с одноканальным УСО подобно работе системы TDMA.

В конечном счете, каждая радиочастотная метка будет передавать ответ по своему выбранному каналу, в то время как все другие радиочастотные метки будут передавать ответы по другим каналам.

6.2.7.3.2.2 Случайная блокировка

В режиме случайной блокировки ответа при случайном выборе канала радиочастотные метки случайным образом принимают решение о передаче или блокировке своего ответа. Если принято решение о передаче ответа, радиочастотная метка выбирает канал ответа случайным образом. Этот режим может использоваться многоканальными УСО для большого количества радиочастотных меток.

Отношение количества блокированных ответов к общему количеству возможных ответов (коэффициент блокировок) может варьироваться командой УСО между 1/2 и 511/512. Это отношение должно увеличиваться с ростом количества меток. Коэффициент блокировок регулирует среднее количество меток, принимающих решение о передаче в течение периода ответа. Этот метод позволяет производить идентификацию нескольких тысяч меток, одновременно находящихся в поле УСО. Коэффициент блокировок помогает уменьшить среднее количество передающих меток до контролируемого уровня.

6.2.7.3.2.3 Режим полной блокировки

УСО может перевести радиочастотную метку в режим полной блокировки ответа (режим ожидания). В этом режиме радочастотная метка не будет отвечать на команды от того же УСО и поэтому не будет конфликтовать с другими радиочастотными метками. Этот режим может использоваться для работы с большим количеством меток, для повышения общего уровня их идентифицируемости. Радиочастотная метка выйдет из режима ожидания, когда она попадет в поле другого УСО.

Радиочастотная метка определяет, что она находится в поле нового УСО на основе данных, содержащихся в его команде, или в случае если радиочастотная метка обнаруживает, что была выключена дольше определенного периода времени.

6.2.7.4 Генератор случайных чисел (RNG)

Для того чтобы генерировать случайные числа при выборе каналов ответа и при блокировке ответов, радиочастотная метка использует генератор случайных чисел (RNG), представляющий собой линейный сдвиговый регистр максимальной длиной 32 бита или более.

Для того чтобы предотвратить появление чрезмерно долгих интервалов молчания, которые могут случаться при работе с таким генератором, как PRBS*, радиочастотная метка может иметь средства для принудительного незаблокированного ответа после определенного числа заблокированных ответов.

- Когда отношение блокировки составляет 1/2, максимальное число заблокированных ответов должно быть 3.
- Когда отношение блокировки составляет 3/4, максимальное число блокированных ответов должно быть 7.
- Когда отношение блокировки составляет 7/8, максимальное число блокированных ответов должно быть 15.
- Когда отношение блокировки составляет 31/32, максимальное число блокированных ответов должно быть 63.

Для простых одноканальных УСО предпочтителен канал G. Для того чтобы такое УСО могло идентифицировать несколько радиочастотных меток, радиочастотные метки должны работать на случайном канале в одном из режимов: «Свободного канала» или «Случайной блокировки» ответа. Чтобы предотвратить чрезмерно длительные периоды без передачи на канале G, радиочастотная метка может иметь средства для принудительной посылки ответа на канале G после последовательности из 15 незаблокированных ответов, которые произошли на каналах, отличных от канала G.

Реализации такого счетчика или его эквивалента, применяемые для принудительного ответа, должны использовать устройство хранения данных, такое как TRAM или эквивалент, чтобы сохранить значения во время коротких периодов отключения питания, см. 6.2.5.23.4.

6.2.7.5 Параметры команды

См. 6.2.5.16.2.

6.2.7.6 Обработка запросов радиочастотной меткой

Радиочастотные метки отвечают только на действительные команды. Функционал радиочастотной метки полностью определяется командой, см. 6.2.5.16.2. Команда определяет поведение метки при выборе канала и блокировке ответа. Радиочастотные метки будут случайным образом выбирать блокировку ответа и канал ответа, как это задано командой. Случайный выбор реализуется с помощью PRBS генератора, встроенного в микросхему радиочастотной метки. Как только радиочастотная метка переходит в режим ожидания, она перестает отвечать на дальнейшие нормальные команды.

6.2.7.7 Описание метода работы системы управления коллизиями

В системе РЧИ идентификация множества меток реализуется с использованием комбинации методов Множественного доступа с частотным и временным разделением каналов (FTDMA).

В ответ на действительную команду каждая радиочастотная метка случайным образом выбирает канал для передачи ответа. На выбранном канале ответ передается один раз. После получения следующей действительной команды каждая радиочастотная метка случайным образом выбирает новый канал и передает ответ, используя новый выбранный канал.

Этот метод скачкообразной перестройки частоты с использованием случайного выбора канала ответа, повторяется для каждой последующей действительной команды.

В дополнение к случайному выбору канала радиочастотная метка может случайным образом блокировать некоторые ответы. Когда происходит блокировка ответа, радиочастотная метка не передает этот ответ. Случайная блокировка необходима при идентификации очень большого количества радиочастотных меток. Как только радиочастотная метка идентифицируется, она на некоторое время переходит в режим ожидания.

Все частотные и временные параметры FTDMA определяются командой. FTDMA обеспечивает лучшую производительность при работе по сравнению с работой на одной частоте в режиме TDMA, потому что множество ответов меток могут быть приняты на различных каналах одновременно.

В случае работы с несколькими метками выгоднее использовать УСО с функцией дуплексной связи. Данные от 8 меток могут быть получены одновременно на 8 каналах, когда УСО специально указывает в командах меткам отвечать на различных каналах.

Для оценки среднего числа меток, которые будут идентифицированы после каждой действительной команды, необходимо применять методы теории вероятностей. Для группы из $n$ меток с $r$ доступ-

---

* *PseudoRandom Binary Sequence — псевдослучайная двоичная последовательность.*

ными каналами и коэффициентом блокировок $m$ среднее число идентифицированных меток $N$ после каждой действительной команда считывания будет определяться по формуле (1):

$$N = n \cdot m \cdot \left(\frac{r-1}{r}\right)^{(n\ \ m\ \ 1)}, \quad (1)$$

где $m$ — вероятность того, что радиочастотная метка передаст ответ.

Наивысшая скорость идентификации достигается, когда число меток, отвечающих одновременно, равно числу доступных каналов. УСО максимизирует скорость идентификации, корректируя коэффициент блокировок так, чтобы выполнялось следующее условие (2):

$$n * m \approx r. \quad (2)$$

Скорость идентификации $N$ как функция количества меток $n$, изменяющегося от 1 до 10 000, для различных коэффициентов блокировок, графически отображается на рисунке 12. При соответствующем выборе коэффициента блокировки скорость идентификации может поддерживаться в диапазоне от 2 до 3 для количества меток до 8 000.

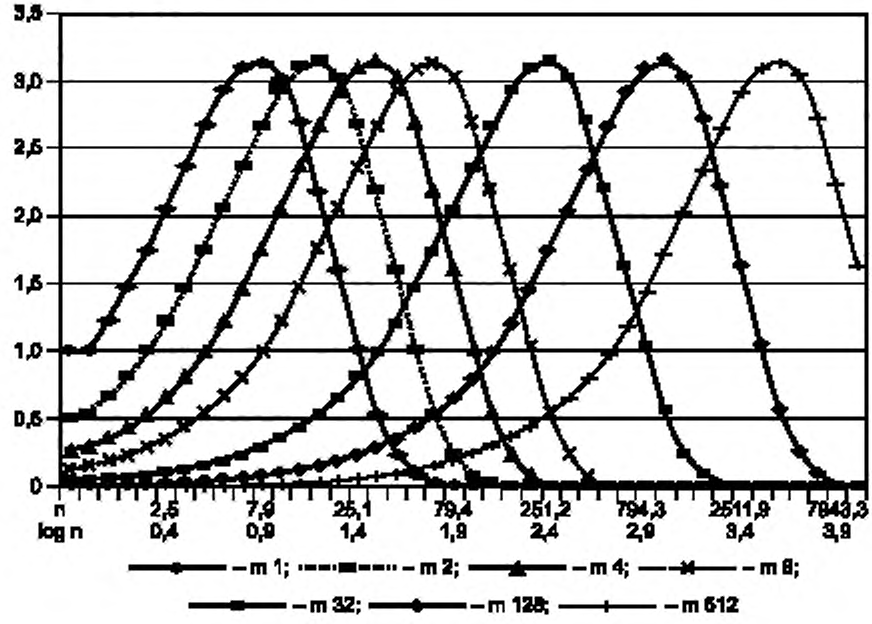



Рисунок 12 — Скорость идентификации

6.2.7.8 Последовательность управления коллизиями для небольшого числа меток

При идентификации и считывании данных небольшого количества меток коэффициент блокировок устанавливается равным 1. Среднее число полученных ответов от меток находится в диапазоне от 1 до 3 на каждую команду считывания. Последовательность действий для идентификации и считывания 8 меток следующая:

a) Радиочастотные метки помещаются в зону считывания УСО.

b) Дается команда считывания с нулевой длиной блока читаемых данных (нулевое считывание).

c) После идентификации меток они временно блокируются (переводятся в режим ожидания).

d) Процесс повторяется, пока все радиочастотные метки не будут идентифицированы.

e) После идентификации данные считываются с помощью специальной команды за одну операцию.

f) Команды объединяются, чтобы данные от 8 меток были считаны одновременно.

Последовательность действий для идентификации 8 радиочастотных меток показана ниже в таблице 23.

Таблица 23 — Идентификация 8 радиочастотных меток

| Действие | Результат | Количество идентифицированных радиочастотных меток |
|---|---|---|
| Старт | — | — |
| УСО передает команду нулевого считывания (на случайном канале) | Все радиочастотные метки отвечают на случайных каналах | 0 |
| УСО получает 3 ответа радиочастотных меток | — | 3 |
| УСО передает специальную команду блокировки для каждой идентифицированной радиочастотной метки | Заблокированные радиочастотные метки находятся в режиме ожидания и временно не будут реагировать на обычные команды | — |
| УСО передает команду нулевого считывании (на случайном канале) | 5 незаблокированных меток отвечают на случайных каналах | 3 |
| УСО получает 3 ответа радиочастотных меток | — | 6 |
| УСО передает специальную команду блокировки для каждой идентифицированной радиочастотной метки | Заблокированные радиочастотные метки находятся в режиме ожидания и временно не будут реагировать на обычные команды | — |
| УСО передает команду нулевого считывания (на случайном канале) | 2 незаблокированные радиочастотные метки отвечают на случайных каналах | — |
| УСО передает специальную команду блокировки для каждой идентифицированной радиочастотной метки | Заблокированные радиочастотные метки находятся в режиме ожидания и временно не будут реагировать на обычные команды | — |
| Конец | Общее время идентификации 8 меток — 5,772 мс (включая минимальное время циклов) | 8 радиочастотных меток проидентифицированы |

После процесса идентификации данные считываются за одну операцию с помощью специальной команды. Считывание данных за одну операцию — самый эффективный по времени метод, хорошо подходит для неизменного, на время чтения, количества меток.

Для каждой команды считывания УСО выбирает определенную метку и неиспользуемый канал для ее ответа (Радиочастотная метка I на канале А и т. д.). Последовательность операций при считывании данных из 8 меток подробно показана ниже в рисунке 13.

Дуплексный обмен данными между УСО и радиочастотными метками позволяет УСО объединять последовательные команды. Это позволяет отвечать 8 радиочастотным меткам одновременно.

Общее время считывания 8 меток — это время 8 команд считывания и единственного ответа от последней радиочастотной метки.

6.2.7.9 Последовательность управления коллизиями для большого числа меток

При идентификации и считывании данных у большого количества меток устанавливаются такой коэффициент блокировок, чтобы среднее количество одновременно отвечающих меток сократилось примерно до количества каналов. Когда правильно выбран коэффициент блокировок, среднее число ответов меток колеблется от 2 до 3 на каждую команду считывания.

Последовательность работы при идентификации 500 меток и считывания 50 слов данных от каждой радиочастотной метки следующая:

a) 500 меток помещаются в поле УСО.

b) Запускается команда нулевого считывания, при этом проверяется число полученных ответов от меток.

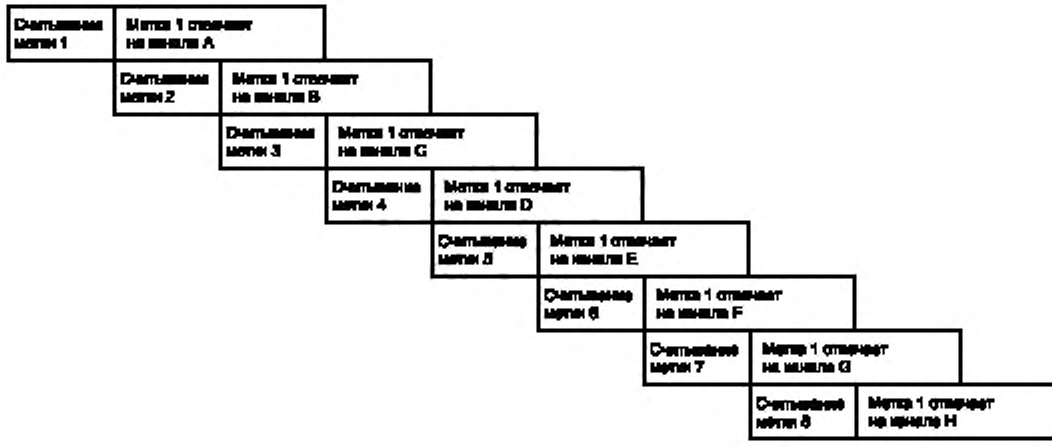



Рисунок 13 — Считывание данных из 8 радиочастотных меток

c) Отношение блокировок увеличивается до тех пор, пока среднее число полученных ответов от меток не будет от 2 до 3 на одно считывание.

d) После того, как радиочастотные метки идентифицируются, они временно блокируются (переводятся в режим ожидания).

e) Последовательность действий повторяется, пока все метки не будут идентифицированы. Коэффициент блокировок корректируется в соответствии с уменьшением числа неидентифицированных меток таким образом, чтобы за один цикл считывания было получено, по крайней мере, 2 ответа от меток.

f) Данные считываются с помощью специальной команды считывания, либо за одну операцию после процесса идентификации, либо непрерывно в ходе процесса идентификации.

g) Команды объединяются так, чтобы данные от 8 меток были считаны одновременно.

Последовательность операций для процесса идентификации подробно показана ниже в таблице 24.

Таблица 24 — Идентификация 500 радиочастотных меток

| Действие | Результат | Число команд и ответов | Число идентифицированных меток |
|---|---|---|---|
| Старт | — | — | — |
| УСО передает команду нулевого считывания (выбор случайного канала) | Все 500 меток отвечают на случайных каналах | — | 0 |
| Коэффициент блокировок в последовательности команд увеличивается до m = 1/128 | 3,9 (в среднем) меток отвечают на случайных каналах | 6 считываний и 6 коротких ответов | — |
| УСО передает команду считывания нулевой длины с установленным коэффициентом блокировок 1/128 | Идентифицированы 175 меток | 70 считываний, 70 коротких ответов, 175 блокировок | 175 |
| УСО передает команду считывания нулевой длины с установленным коэффициентом блокировок 1/32 | Идентифицированы 245 меток | 98 считываний, 98 коротких ответов, 245 блокировок | 420 |

*Окончание таблицы 24*

| Действие | Результат | Число команд и ответов | Число идентифицированных меток |
|---|---|---|---|
| УСО передает команду считывания нулевой длины с установленным коэффициентом блокировок 1/8 | Идентифицированы 60 меток | 24 считывания, 24 коротких ответа, 60 блокировок | 480 |
| УСО передает команду считывания нулевой длины с установленным коэффициентом блокировок ¼ | Идентифицированы 15 меток | 6 считываний, 6 коротких ответов, 15 блокировок | 495 |
| УСО передает команду считывания данных нулевой длины с установленным коэффициентом блокировок 1/2 | Идентифицированы 2 радиочастотные метки | 1 считывание, 1 короткий ответ, 2 блокировки | 497 |
| УСО передает команду считывания данных нулевой длины с установленным коэффициентом блокировок 1 | Идентифицированы 2 радиочастотные метки | 1 считывание, 1 короткий ответ, 2 блокировок | 499 |
| УСО передает команду считывания данных нулевой длины с установленным коэффициентом блокировок 1 | Идентифицирована 1 радиочастотная метка | 1 считывание, 1 короткий ответ, 1 блокировка | 500 |
| Конец и итоги | Общее время идентификации 500 меток меньше 0,390 с (включая минимальное время циклов приема-передачи) | | Идентифицированы 500 меток |

Данные могут быть считаны за одну операцию после процесса идентификации или в рамках непрерывного процесса во время идентификации. Считывание данных за одну операцию — самый эффективный по времени метод, хорошо подходит для неизменного, на время считывания, количества меток. Считывание данных меток непрерывно во время процесса идентификации менее эффективно, но хорошо подходит для динамично изменяющегося количества меток.

Последовательность действий при считывании данных за одну операцию подробно показана в таблице 25.

Таблица 25 — Считывание данных 500 статических меток

| Действие | Результат | Количество команд и ответов | Считанные радиочастотные метки |
|---|---|---|---|
| Старт | — | — | — |
| Специальная команда считывания (50 слов) на каналах от А до Н | Считывание данных из радиочастотной метки 1 на канале А, из радиочастотной метки 2 на канале В, из радиочастотной метки 8 на канале Н | 8 считываний, 1 ответ считывания | 8 |
| Ожидание последних 264 мкс ответа радиочастотной метки на канале А | Радиочастотная метка 9 готова к приему команды считывания и начнет отвечать на канале А после окончания ответа радиочастотной метки 1 | — | — |

*Окончание таблицы 25*

| Действие | Результат | Количество команд и ответов | Считанные радиочастотные метки |
|---|---|---|---|
| Специальная команда считывания (50 слов) на каналах от А до Н | Считывание данных из радиочастотной метки 9, на канале А, из радиочастотной метки 10 на канале В, из радиочастотной метки 16 на канале Н | 1 ответ считывания | 16 |
| Последовательность операции продолжается, пока все радиочастотные метки не будут считаны | Последовательность повторяется в общей сложности 63 раза | 61 ответ считывания | 500 |
| Завершение и результат | Общее время считывания 500 меток — 0,540 с | — | Считано 500 статических меток |

Общее время идентификации и считывания 50 слов данных из 500 меток за одну операцию составляет меньше чем 0,390 с + 0,540 с = 0,930 с.

Последовательность операций для непрерывного процесса считывания данных подробно описана в таблице 26. Для непрерывного процесса считывания отдельная операция блокировки, приведенная в последовательности действий при идентификации, не требуется, она необходима как часть последовательности действий при считывании.

Таблица 26 — Считывание данных из 500 динамических меток

| Действие | Результат | Количество команд и ответов | Считанные радиочастотные метки |
|---|---|---|---|
| Старт | — | — | — |
| УСО идентифицирует 8 меток | 8 радиочастотных меток готовы к считыванию данных | — | — |
| Специальная команда считывания (50 слов) на каналах от А до Н | Считывание данных из радиочастотной метки 1 на канале А, радиочастотной метки 2 — на канале В, радиочастотной метки 8 — на канале Н | 8 считываний, 1 ответ считывания | 8 |
| Блокировка меток после получения данных* | — | 1 блокировка | — |
| Повторение последовательности для каждых 8 идентифицируемых меток | Последовательность повторяется в общей сложности 63 раза | 492 считывания, 62 ответа считывании, 62 блокировки | 500 |
| Завершение и результат | Общее время считывания 500 динамических меток — 0,686 с | — | 500 динамических меток считаны |

Общее время идентификации и считывания 50 слов данных из 500 меток в рамках непрерывного процесса составляет менее чем 0,258 с + 0,686 с = 0,944 с.

6.2.7.10 Определение временных параметров

Процесс идентификации и считывания множества меток требует использования команд считывания данных с нулевой длиной блока и ответов на них, команд перевода в режим ожидания, команд считывания и ответов на них. Временные параметры команд и ответов представлены в таблице 27.

* *Перевод меток в режим ожидания.*

Таблица 27 — Временные параметры команд и ответов при идентификации множества меток

| Тип Команды/Ответа | Описание | Время, мкс |
|---|---|---|
| Команда нулевого считывания | Короткая команда считывания данных нулевой длины используется для идентификации | 264 |
| Ответ при нулевом чтении | Короткий ответ с данными нулевой длины | 906 |
| Команда блокировки | Временная блокировка определенной радиочастотной метки, идентифицированной в процессе считывании | 264 |
| Команда считывания (*W* слов) | Команда считывания для *W* слов данных | 264 |
| Ответ при считывании (*W* слов) | Ответ при считывании *W* слов данных | 906 + 151*W* |

**6.2.8 Порядок следования меток**

Для приложений, требующих определения порядка следования предметов с целью их сортировки, система РЧИ должна однозначно коррелировать каждую радиочастотную метку с каждым предметом вне зависимости от наличия множества других меток и расстояния между радиочастотными метками. Такая возможность представлена в МОДА 2.

Например, на конвейере со скоростью ленты 3,6 м/с, расстоянием между радиочастотными метками 15 см, в режиме переключения направления излучения УСО, определение порядка следования меток должно происходить менее чем за 13,9 мс. В системах, в которых имеется задержка из-за процесса идентификации меток, не хватает времени для определения порядка следования меток.

В этой ситуации порядок следования меток определяется путем определения моментов времени появления меток в поле УСО. Все команды имеют отметки времени, и радиочастотные метки хранят первую отметку времени, полученную при появлении радиочастотной метки в поле УСО. Сохраненная метка времени передается во всех ответах радиочастотной метки. Точность метки времени находится на уровне миллисекунд. Таким образом, определение порядка следования меток не зависит от скорости идентификации.

Для сохранения значений меток времени во время коротких отключений питания должна использоваться Память для временного хранения данных с произвольным доступом (TRAM), см. 6.2.5.23.4.

**6.2.9 Команды**

Все команды описаны в вышеприведенных подразделах.

**6.2.10 Прикладной уровень радиоинтерфейса**

Прикладной уровень должен быть определен и управляться исключительно УСО и не должен быть перенесен на радиоинтерфейс.

**6.2.11 Дополнительная функциональность**

УСО и радиочастотная метка могут поддерживать дополнительную функциональность.

6.2.11.1 Режим кодирования «1 из 4»

В дополнение к кодированию MFM, команды могут быть закодированы в режиме «1 из 4». Режим «1 из 4» допускает большую глубину модуляции, по сравнению с MFM, при сохранении той же спектральной маски передатчика и обеспечении скорости передачи данных 423,75 Кбит/с (fc/32). Для режима «1 из 4» изменения фазы (PJM) будут между ±3,0° и ±4,0°.

6.2.11.2 Кодирование данных при использовании режима «1 из 4»

Режим «1 из 4» кодирует битовые пары путем использования позиции символа в двухбитовом периоде кадра. Для этого протокола:

- Закодированный символ передается в виде одной смены фазы PJM силового питающего поля УСО,
- Режим «1 из 4» может быть немного изменен таким образом, что отсутствие переданного символа в двухбитовом периоде кадра означает передачу 00, см. ниже;
- Передача данных начинается с младшего значащего бита пары.

Кодирование битовых пар для режима 1 из 4 показано на рисунке 14.

Обычно фронты сигналов, показанные на рисунке 14 и на рисунке 15, должны быть синхронизированы с силовым питающим полем. Если эти фронты не синхронизированы, тогда они будут сгенерированы УСО в пределах ±0,04 мкс относительно длительностей интервалов времени, показанных на рисунке 14.

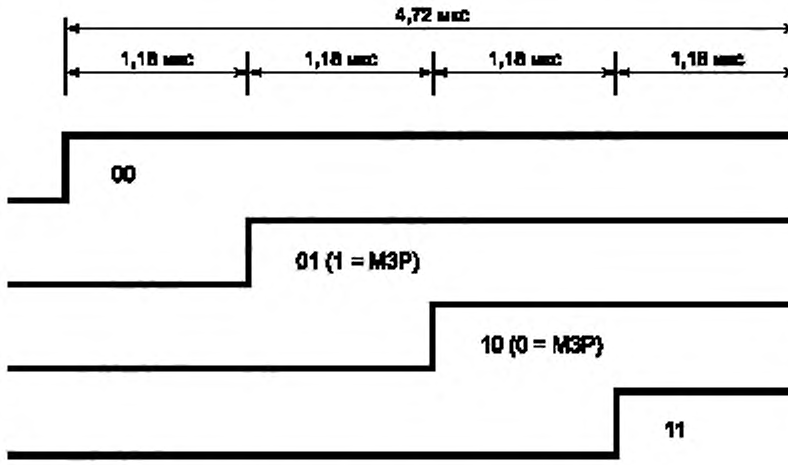



Рисунок 14 — Режим кодирования «1 из 4»

Для этого протокола, если текущие передаваемые данные битовой пары должны быть декодированы как 00, и в предыдущем кадре присутствовал символ, тогда никакого символа больше передавать не нужно. Если символ не передается, боковые полосы частот передатчика могут быть сужены. Этот метод гарантирует периодическую передачу символов в длинных последовательностях 00 данных, а также предотвращает передачу коротких импульсов, которые случаются при передаче данных 0011.

Пример команды кодирования двоичной последовательности 11100001 методом «1 из 4» показан на рисунке 15.




Рисунок 15 — Пример кодирования методом «1 из 4»

6.2.11.3 Кодирование битовых пар для режима «1 из 4»

Флаг команды определяет начало команды и временные диаграммы в соответствии с битовыми интервалами. Флаг режима «1 из 4» отличается от флага MFM, это позволяет на ходу декодировать команды MFM или команды в режиме «1 из 4» без какого-либо конфигурирования. Флаг состоит из трех частей:

- синхронизирующая строка из 6 бит данных представленных в формате MFM;
- нарушение кодирования для режима «1 из 4», не допустимое в обычной передаче в MFM или в «1 из 4». Нарушение состоит из 6 последовательных изменений состояния, разделенных интервалом: в 1 бит, в 2 бита, в 2 бита, в 1,5 бита и в 2 бита. Конец шестого (последнего) изменения определяет начало битового интервала;
- оставшийся 0 (ноль) MFM определяет конец флага и начало команды.

Синхронизирующая строка, нарушение кодирования и конечный ноль для двух возможных флагов команды показаны на рисунке 16, а также временные диаграммы для двух возможных флагов команды.

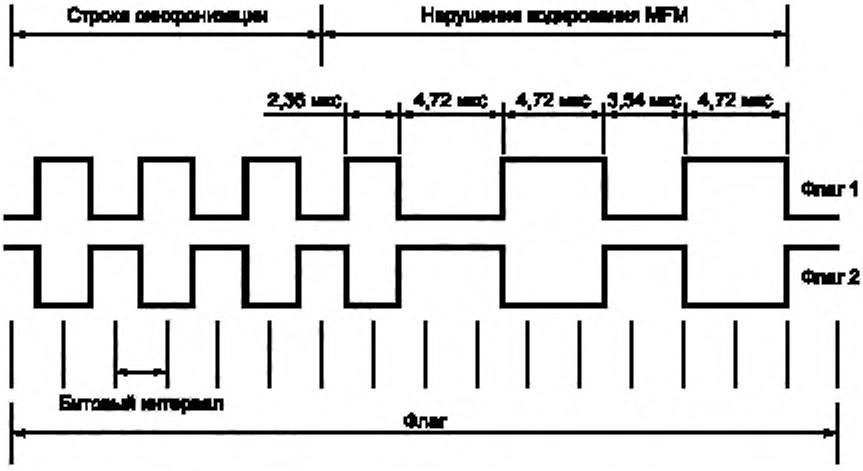



Рисунок 16 — Кодирование и разбивка по времени (согласование по времени) для двух возможных флагов команды в режиме 1 из 4

### 6.3 МОДА 3: физический уровень, система управления коллизиями и протоколы МОДА 3

МОДА 3 не совместим с любыми другими МОДА, определенными в настоящем стандарте.

МОДА 3 не создает помехи любым другим МОДА, определенным в настоящем стандарте.

В приложении L приводится краткая информация о функциях, доступных для меток, которые соответствуют МОДА 3.

#### 6.3.1 Обзор протокола

6.3.1.1 Физический уровень

УСО передает информацию одной или более радиочастотным меткам путем модуляции несущей РЧ, используя двухполосную амплитудную манипуляцию (ASK) с использованием время-импульсного кодирования (PIE). Радиочастотные метки получают энергию для работы от этой же модулированной несущей РЧ.

УСО получает информацию от радиочастотной метки путем передачи немодулированной несущей РЧ и приема модулированного ответного сигнала. Радиочастотные метки передают информацию, модулируя амплитуду и/или фазу радиочастотного сигнала. Формат кодирования, который выбирается в ответ на команды УСО, представляет собой модулирование поднесущей или основной частоты FM0 с использованием либо Манчестерского кодирования, либо кода Миллера. Каналы соединения между УСО и радиочастотными метками полудуплексные, это означает, что радиочастотной метке не требуется демодулировать команды УСО во время собственного процесса модуляции. Радиочастотная метка не должна отвечать на обязательные или дополнительные команды, используя дуплексную связь.

УСО, как и радиочастотные метки, могут дополнительно реализовать физический уровень с помощью фазовой модуляции колебаний (PJM), используя MFM (модифицированную частотную модуляцию) при передаче от УСО к радиочастотной метке и MFM кодирование с двоичной фазовой манипуляцией (BPSK) поднесущей при передаче от радиочастотной метки к УСО.

6.3.1.2 Уровень идентификации радиочастотной метки

УСО управляет группами меток, используя три основные операции.

**a) Выбор.** Операция выбора группы меток для осуществления инвентаризации и доступа. Команда *Select* может успешно использоваться для выбора определенной группы меток на основе заданных пользователем критериев. Настоящая операция аналогична выбору записей из базы данных.

**b) Инвентаризация.** Операция идентификации радиочастотных меток. УСО начинает инвентаризацию путем передачи команды *BeginRound* (Начать цикл опроса) за один или два сеанса. Ответить могут одна или более меток. УСО, получает единственный ответ и запрашивает у метки *PC/слово(слова) XPC, UII,* и, при необходимости, CRC-16 пакета. Инвентаризация включает в себя несколько команд. Цикл инвентаризации выполняется во время одного и только одного сеанса.

**c) Доступ.** Операция соединения с радиочастотной меткой (считывание и/или запись). Индивидуальная метка должна быть однозначно идентифицирована для получения доступа. Доступ включает в себя несколько команд, некоторые из которых могут обеспечить защиту данных с помощью шифра Вернама (*One-time pad*) при передаче данных от радиочастотной метки к УСО (R=>T).

**6.3.2 Общие сведения**

6.3.2.1 Функциональная совместимость для МОДА 3

МОДА 3 работает на частоте 13,56 МГц.

МОДА 3 не совместим с МОДА 1 и МОДА 2, определенными в настоящем стандарте, но все три режима могут работать одновременно в одной и той же среде.

**6.3.3 Физический уровень, система управления коллизиями и протоколы**

6.3.3.1 Нормативные аспекты: физический уровень и уровень управления доступом к среде (MAC)

Таблица 28 и таблица 29 представляют обзор параметров связи для R => T и T => R в соответствии с настоящим стандартом; для получения их подробного описания см. соответствующие подразделы. Те параметры, которые не применяются или не используются в данной спецификации, обозначены «N/A», что указывает на то, что этот параметр «не применим». Таблица соответствует *[1]*.

Таблица 28 — Каналы связи «УСО — радиочастотная метка»

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Int: 1 | Диапазон рабочих частот | Фиксированная частота см. Int: 1A | 6.3.3.3.1 |
| M3-Int: 1a | Рабочая частота по умолчанию | 13,56 МГц | 6.3.3.3.1 |
| M3-Int: 1b | Рабочие каналы (широкополосные системы) | N/A | N/A |
| M3-Int: 1c | Точность рабочей частоты | В соответствии с национальными требованиями | 6.3.3.3.1.2.1 |
| M3-Int: 1d | Частота перестройки (для систем, использующих алгоритм псевдослучайной перестройки частоты [FHSS]) | N/A | N/A |
| M3-Int: 1e | Последовательность частот (для систем, использующих алгоритм псевдослучайной перестройки частоты [FHSS]) | N/A | N/A |
| M3-Int: 2 | Полоса пропускания используемого канала | N/A | N/A |
| M3-Int: 3 | Максимальная напряженность магнитного поля, УСО<br><br>Предельные значения напряженности магнитного поля в коммуникационной зоне | УСО не должно создавать поле напряженностью более 12 А/м в любой части объема, где может находиться радиочастотная метка, размер которой соответствует размеру идентификационной карты по ИСО.<br>Максимальная рабочая напряженность поля:<br>5 А/м для радиочастотных меток, размер которых соответствует размеру идентификационной карты по *[4]*. Методы испытаний определены в *ГОСТ Р 56914*. Для меток других размеров изготовитель меток должен указать максимальную рабочую напряженность поля* | N/A |
| M3-Int: 4 | Паразитное излучение УСО | В соответствии с национальными требованиями использования радиочастотного спектра | 6.3.3.3.1.2.5 |

* *Решением ГКРЧ 07-20-03-001 определена максимальная напряженность магнитного поля на расстоянии 10 м от УСО, которая составляет 60 дБ(мкА/м).*

*Продолжение таблицы 28*

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Int: 4a | Паразитное излучение УСО внутри рабочего диапазона частот (широкополосные системы) | В соответствии с национальными требованиями использования радиочастотного спектра | 6.3.3.3.1.2.5 |
| M3-Int: 4b | Паразитное излучение УСО вне рабочего диапазона частот | В соответствии с национальными требованиями использования радиочастотного спектра | 6.3.3.3.1.2.5 |
| M3-Int: 5 | Спектральная маска передатчика | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Int: 6 | Временные диаграммы | См. ниже | 6.3.3.3.1.6 |
| M3-Int: 6a | Время цикла передача-прием | Менее 73,1 мкс, максимальное (1024/Fc—32/Fc) | 6.3.3.3.1.6, рисунок 40 и таблица 41 |
| M3-Int: 6b | Время цикла прием-передача | При соединении с радиочастотной меткой, минимум — 151 мкс; максимум — 1208 мкс, когда радиочастотная метка находится в состоянии Reply & Acknowledged; в иных случаях — без максимального предела | 6.3.3.3.1.6, рисунок 40 и таблица 41 |
| M3-Int: 6c | Время задержки или линейно нарастающий сигнал УСО | 1500 мкс, максимальное время установления | 6.3.3.3.1.2.6, таблица 34 |
| M3-Int: 6d | Время затухания или линейно падающий сигнал УСО | 500 мкс, максимум | 6.3.3.3.1.2.7, таблица 35 |
| M3-Int: 7 | Модуляция | Метод ASK: минимум 10 %, максимум 30 %.<br>Дополнительный метод PJM: минимальное отклонение ±3,0°, максимальное отклонение ±6,0° | 6.3.3.3.1.2.2 |
| M3-Int: 7a | Последовательность значений частоты (системы широкополосной модуляции с прямым расширением спектра [DSSS]) | N/A | N/A |
| M3-Int: 7b | Тактовая частота (широкополосные системы) | N/A | N/A |
| M3-Int: 7c | Точность значений тактовой частоты (широкополосные системы) | N/A | N/A |
| M3-Int: 7d | Индекс модуляции | Метод ASK: (A-B)/(A + B) коэффициент мин 10 % макс 30 % | 6.3.3.3.1.2.5, рисунок 19, таблица 32 |
| M3-Int: 7e | Рабочий цикл | Как указано | 6.3.3.3.1.2.5 |
| M3-Int: 7f | Девиация частотной модуляции | N/A | N/A |
| M3-Int: 8 | Кодирование данных | Метод ASK: PIE<br>Дополнительный метод PJM: MFM | 6.3.3.3.1.2.3, рисунок 17, рисунок 18 |
| M3-Int: 9 | Скорость передачи данных в битах | Метод ASK: от 26,7 Кбит/с до 100 Кбит/с (при условии равновероятных данных)<br>Дополнительный режим PJM: 212 Кбит/с | 6.3.3.3.1.2.4 и опционально 6.3.3.3.1.2 |
| M3-Int: 9a | Точность скорости передачи данных в битах | ±1 % минимум | 6.3.3.3.1.2.3 |

*Окончание таблицы 28*

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Int: 10 | Точность модуляции УСО | Как указано | 6.3.3.3.1.2.3 |
| M3-Int: 11 | Заголовок | Требуется | 6.3.3.3.1.2.8 |
| M3-Int: 11a | Длина заголовка | Как указано | 6.3.3.3.1.2.8 |
| M3-Int: 11b | Сигнал(ы) заголовка | Как указано | Рисунок 22, рисунок 23 |
| M3-Int: 11c | Последовательность битов синхронизации | Нет | N/A |
| M3-Int: 11d | Последовательность синхронизации кадров | Требуется | 6.3.3.3.1.2.8 |
| M3-Int: 12 | Скремблирование (системы с расширенным спектром) | N/A | N/A |
| M3-Int: 13 | Порядок передачи битов | Наиболее значащий бит передается первым | 6.3.3.3.1.4 |
| M3-Int: 14 | Процесс активации | Как указано | 6.3.3.3.1.2.6 |
| M3-Int: 15 | Поляризация | N/A | N/A |

Таблица 29 — Каналы связи «радиочастотная метка-УСО»

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Tag: 1 | Диапазон рабочих частот | 13,56 МГц + /– поднесущие частоты радиочастотной метки, как указано в: 7e | 6.3.3.3.1.1, таблица 36 и таблица 37 |
| M3-Tag: 1a | Рабочая частота по умолчанию | Фиксированная несущая частота, как указано в Int: 1a | 6.3.3.3.1.1 |
| M3-Tag: 1b | Рабочие каналы (для широкополосных систем) | N/A | N/A |
| M3-Tag: 1c | Точность рабочей частоты | Как указано | 6.3.3.3.1.2.1 |
| M3-Tag: 1d | Частота перестройки (системы использующие алгоритм псевдослучайной перестройки частоты [FHSS]) | N/A | N/A |
| M3-Tag: 1e | Последовательность скачкообразной перестройки частоты ответа меток на запрос УСО (системы использующие алгоритм Псевдослучайной перестройки частоты [FHSS]) | N/A | N/A |
| M3-Tag: 2 | Ширина полосы пропускания используемого канала | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Tag: 3 | Максимальная напряженность магнитного поля | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Tag: 4 | Паразитное излучение передатчика | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |

*Продолжение таблицы 29*

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Tag: 4a | Паразитное излучение внутри полосы частот (для широкополосных систем) | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Tag: 4b | Паразитное излучение вне полосы частот | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Tag: 5 | Спектральная маска передатчика | В соответствии с национальными требованиями использования радиочастотного спектра | N/A |
| M3-Tag: 6 | Временные диаграммы | См. ниже | 6.3.3.3.1.6 |
| M3-Tag: 6a | Время цикла передача-прием | Менее 151 мкс | 6.3.3.3.1.6, рисунок 40 и таблица 41 |
| M3-Tag: 6b | Время цикла прием-передача | 75,5 мкс номинальное (1024/Fc) | 6.3.3.3.1.6, рисунок 40 и таблица 41 |
| M3-Tag: 6c | Время задержки или линейно нарастающий сигнал | Готовность к приему команд менее чем за 1500 мкс | 6.3.3.3.1.2.6 |
| M3-Tag: 6d | Время затухания или линейно падающий сигнал | N/A | N/A |
| M3-Tag: 7 | Модуляция | Метод ASK: путем модуляции нагрузкой | 6.3.3.3.1.3.1 |
| M3-Tag: 7a | Последовательность значений частоты (для систем с прямым расширением спектра [DSSS]) | N/A | N/A |
| M3-Tag: 7b | Тактовая частота (для широкополосных систем) | N/A | N/A |
| M3-Tag: 7c | Точность значений тактовой частоты (для широкополосных систем) | N/A | N/A |
| M3-Tag: 7d | Отношение ON/OFF (времени излучения к времени ожидания) | Зависит от радиочастотной метки, не определено в этом документе | N/A |
| M3-Tag: 7e | Частота поднесущих | Метод ASK: 423 кГц (fc/32) или 847 кГц (fc/16) (выбирается УСО)<br>Дополнительный метод PJM:<br>Канал A 969 кГц (деление на 14)<br>Канал B 1233 кГц (деление на 11)<br>Канал C 1507 кГц (деление на 9)<br>Канал D 1808 кГц (деление на 7,5)<br>Канал E 2086 кГц (деление на 6,5)<br>Канал F 2465 кГц (деление на 5,5)<br>Канал G 2712 кГц (деление на 5)<br>Канал H 3013 кГц (деление на 4,5)<br>для ответов меток | 6.3.3.3.1.3.11, таблица 36 и опционально: 6.3.3.3.1.3.11, таблица 38 |
| M3-Tag: 7f | Точность частоты поднесущей | Соответствует точности несущей частоты | 6.3.3.3.1.3.11, таблица 36 |

*Окончание таблицы 29*

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| M3-Tag: 7g | Модуляция поднесущей | Метод ASK: Манчестерское кодирование или код Миллера,<br>см. Tag: 9 и таблицу 35<br>Опциональный метод PJM:<br>BPSK при 106 Кбит/с | 6.3.3.3.1.3.5, 6.3.3.3.1.3.7 и опционально 6.3.3.3.1.3.9 |
| M3-Tag: 7h | Рабочий цикл | FM0: 50 %, номинальное<br>Поднесущая: 50 %, номинальное | 6.3.3.3.1.3.3, 6.3.3.3.1.3.5 |
| M3-Tag: 7i | Девиация ЧМ | N/A | N/A |
| M3-Tag: 8 | Кодирование данных | Метод ASK: полоса частот FM0 или поднесущая, модулированная с помощью методов Манчестерского кодирования или кода Миллера — (выбирается УСО)<br>Опциональный метод PJM: MFM для ответов | 6.3.3.3.1.3.2 |
| M3-Tag: 9 | Скорость передачи данных в битах | Метод ASK:<br>FM0, 424 Кбит/сек или 848 Кбит/с;<br>Модулирование поднесущей, от 53 Кбит/с до 212 Кбит/сек<br>Опциональный метод PJM: 106 Кбит/с на каждом канале ответа | 6.3.3.3.1.3.11, таблица 37 и опционально таблица 38 |
| M3-Tag: 9a | Точность скорости передачи данных в битах | Соответствует точности поднесущей частоты, см. Tег: 7f | N/A |
| M3-Tag: 10 | Точность модуляции передатчика радиочастотной метки (системы, использующие псевдослучайную перестройку рабочей частоты [FHSS]) | N/A | N/A |
| M3-Tag: 11 | Заголовок | Требуется | 6.3.3.3.1.3.2 |
| M3-Tag: 11a | Длина заголовка | Как указано | 6.3.3.3.1.3.2 |
| M3-Tag: 11b | Форма сигнала заголовка | Как указано | 6.3.3.3.1.3.2 |
| M3-Tag: 11c | Последовательность битов синхронизации | Нет | N/A |
| M3-Tag: 11d | Последовательность синхронизации кадра | Метод ASK: Нет<br>Дополнительный метод PJM: как указано для ответов | N/A и 6.3.3.3.1.3.10 |
| M3-Tag: 12 | Скремблирование (для систем с расширенным спектром) | N/A | N/A |
| M3-Tag: 13 | Порядок передачи битов | Старший значащий бит первым | 6.3.3.3.1.4 |
| M3-Tag: 14 | Резервный | Намеренно оставлен пустым | N/A |
| M3-Tag: 15 | Поляризация | N/A | N/A |
| M3-Tag: 16 | Минимальная полоса пропускания приемника радиочастотной метки | N/A | N/A |

### 6.3.3.2 Параметры логических операций

Таблица 30 и таблица 31 идентифицируют и описывают параметры, используемые УСО во время процедуры выбора, инвентаризации и доступа к радиочастотным меткам согласно этой спецификации. Те параметры, которые не применяются или не используются в данной спецификации, обозначены «N/A», что указывает на то, что этот параметр «не применим».

Таблица 30 — Параметры инвентаризации и доступа к радиочастотной метке

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| P:1 | «Кто передает первым» | УСО | 6.3.3.3 |
| P:2 | Возможность адресации радиочастотной метки | Как определено | 6.3.3.3.4.1 |
| P:3 | *UII* метки | Содержится в памяти радиочастотной метки | 6.3.3.3.4.1.2 |
| P:3a | *Длина UII* | Как определено | 6.3.3.3.4.1.2 |
| P:3 | *UII* метки | Содержится в памяти радиочастотной метки | 6.3.3.3.4.1.2 |
| P:3a | Длина *UII* | Как определено | 6.3.3.3.4.1.2 |
| P:3b | Формат *UII* | NSI <$100_h$: Как указано в *[2]*<br>NSI> = $100_h$: Как указано*[8]* | 6.3.3.4.1.2.2, 6.3.3.4.1.2.3 и 6.3.3.4.1.2.4 |
| P:4 | Размер при считывании | Кратно 16-битам | 6.3.3.4.11.3.2, таблица 62 |
| P:5 | Размер записи | Кратно 16-бит | 6.3.3.4.11.3.3, таблица 63, рисунок 46, таблица 75 |
| P:6 | Время транзакции считывания | Меняется в зависимости от скорости соединения R => T и T => R и количества считываемых битов | 6.3.3.4.11.3.2 |
| P:7 | Время транзакции записи | 20 мс (максимум) после завершения команды *Записи* | 6.3.3.4.11.3.3, рисунок 46 |
| P:8 | Обнаружение ошибок | УСО->радиочастотная метка:<br>CRC-5 или CRC-16, как это определено в соответствующих подразделах<br>Радиочастотная метка->УСО:<br>CRC-5 или CRC-16, как это определено в соответствующих положениях | 6.3.3.3.1.5 |
| P:9 | Исправление ошибок | N/A | N/A |
| P:10 | Размер памяти | Зависит от метки | N/A |
| P:11 | Структура команды и возможности расширения | Как определено | Таблица 46 |

Таблица 31 — Параметры системы управления коллизиями

| Позиция | Название параметра | Описание | Подраздел |
|---|---|---|---|
| A:1 | Тип (вероятностный или детерминированный) | Вероятностный | 6.3.3.4.6 |
| A:2 | Линейность | Линейный до $2^{15}$ меток в поле УСО | 6.3.3.4.8 |
| A:3 | Емкость инвентаризации меток | >$2^{15}$ меток | 6.3.3.4.8 |

6.3.3.3 Описание работы

Рабочий процесс определяет физические и логические требования для метода «УСО передает первым» (ITF), случайно-слотового антиколизионного алгоритма, системы РЧИ (радиочастотной идентификации), работающей на частоте 13,56 МГц.

Настоящая спецификация описывает два метода работы:

- Метод ASK — (Обязательный). УСО должен связаться с одной или более радиочастотными метками, используя ASK модуляцию, PIE-кодирование. Радиочастотная метка должна ответить на команды ASK УСО, используя модуляцию канала соединения «радиочастотная метка-УСО», определенную в таблице 36 в пункте 6.3.3.3.1.3.11 и в таблице 37. Радиочастотная метка не должна изменять вид модуляции и скорость передачи данных в битах.

- Метод PJM — (Опциональный). УСО может в качестве опции связываться с одной или более радиочастотными метками, используя PJM-модуляцию, MFM-кодирование. Радиочастотные метки, которые поддерживают Метод PJM, должны отвечать на команды УСО, переданные с помощью метода PJM, используя канал соединения «радиочастотная метка-УСО», определенный в пункте 6.3.3.3.1.3.11 и в таблице 38. Радиочастотные метки, которые не поддерживают Метод PJM, не должны отвечать. Радиочастотная метка не должна изменять вид модуляции и скорость передачи данных в битах.

Оба метода используют общую структуру памяти и механизм работы с общим набором логических команд.

Примечание — Ссылки на специфические функции или команды ASK или PJM в данном документе обозначаются как Метод ASK и Метод PJM соответственно.

6.3.3.3.1 Радиоинтерфейс связи

Радиоинтерфейс сигнала связи между УСО и радиочастотной меткой может рассматриваться как физический уровень в многоуровневой системе сетевых коммуникаций. Этот интерфейс определяет частоты, модуляцию, кодирование данных, радиосигнал, скорость передачи данных в битах и другие параметры, требуемые для радиочастотной связи.

6.3.3.3.1.1 Рабочие частоты

Радиочастотные метки должны быть способны получать электропитание от УСО и обмениваться данными с УСО на частоте 13,56 МГц.

6.3.3.3.1.2 Канал связи «УСО-радиочастотная метка» (R => T)

Метод ASK (метод амплитудной манипуляции): УСО должен осуществлять связь с одной или несколькими радиочастотными метками путем модуляции несущей, используя ASK с PIE кодированием. УСО должны использовать фиксированные формат модуляции и скорость передачи данных в битах в течение всего цикла инвентаризации, где «цикл инвентаризации» определен в 6.3.3.4.8.

Метод PJM: УСО осуществляют связь с одной или несколькими радиочастотными метками путем модуляции несущей, используя модуляцию PJM с MFM кодированием при фиксированной скорости передачи данных 212 Кбит/с.

6.3.3.3.1.2.1 Точность частоты УСО

Точность частоты УСО должна удовлетворять национальным требованиям использования радиочастотного спектра.

6.3.3.3.1.2.2 Модуляция

Метод ASK (метод амплитудной манипуляции): все УСО должны осуществлять связь, используя метод ASK, подробнее в Приложении J.

Метод PJM (метод фазовой модуляции колебаний): УСО могут поддерживать метод PJM как указано в приложении B.

6.3.3.3.1.2.3 Кодирование данных

Метод ASK: в соединении УСО-радиочастотная метка R =>T должно использоваться PIE кодирование, представленное на рисунке 17. Интервал Tari является интервалом времени связи УСО-радиочастотная метка продолжительностью, соответствующей передаче нулевого бита. Высокие значения представляют передачу незатухающей волны (CW), низкие значения представляют ослабление незатухающей волны. Допуск по всем параметрам должен быть ±1 %, если не указано иное.

Глубина импульсной модуляции, время нарастания, время спада и PW (длительность импульса) должны соответствовать данным, указанным в таблице 32, и должны быть одинаковыми для данных-**0** и данных-**1**. УСО должны использовать фиксированную глубину модуляции, время нарастания, время спада, длительность импульса PW и время интервала Tari в течение всего цикла инвентаризации. Огибающая должна быть в соответствии с рисунком 19.

Метод PJM: в соединении УСО-радиочастотная метка R =>T должна использоваться модуляция PJM с MFM кодированием на скорости 212 Кбит/с. Период битового интервала, используемый для кодирования команды — 4,72 мкс (64 периода несущей на частоте 13,56 МГц). Скорость изменения фазы указана на рисунке 20. УСО должны использовать фиксированное значение изменения фазы в течение всего цикла инвентаризации.

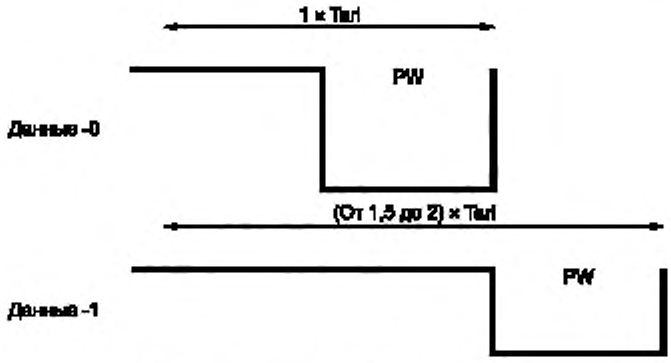



Рисунок 17 — Метод ASK: символы PIE

Значение бита определяется по изменению состояния. Биты кодируются с использованием правил MFM кодирования. Эти правила кодирования определяются следующим образом:

- Данные-1 — значение бита «1» определяется изменением состояния в середине битового интервала.
- Данные-0 — значение бита «0» определяется изменением состояния в начале битового интервала.
- Там, где значение бита данные-0 непосредственно следует за значением бита данные-1 изменения состояния нет.

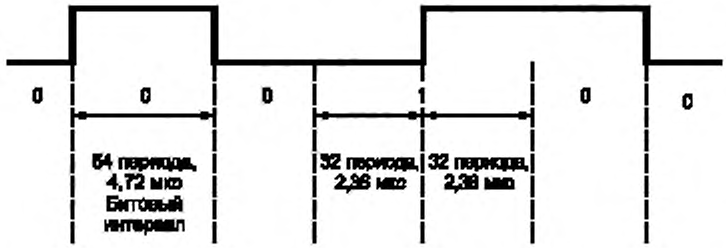



Рисунок 18 — Метод PJM: MFM кодирование команды и временные диаграммы двоичной последовательности 000100

Пример MFM кодирования команды и двоичной последовательности показан на рисунке 18. Фронты сигнала представляют собой небольшие (например, ±3°) фазовые изменения. Как правило, фронты, показанные на рисунке 18, должны быть синхронизированы с несущей частотой. Если не синхронизированы, тогда эти фронты сгенерированы УСО с погрешностью ± один период несущей частоты.

6.3.3.3.1.2.4 Метод ASK: значения интервала Tari

УСО должны связываться, используя значения интервала Tari в диапазоне от 8 мкс до 25 мкс включительно. УСО должен использовать фиксированную длину символов данные-0 и данные-1 в течение всего цикла инвентаризации, «цикл инвентаризации» определен в 6.3.3.4.8. Выбор значения интервала Tari должен производиться в соответствии с национальными требованиями использования радиочастотного спектра.

Примечание — Соответствие требованиям норм оценивается, по крайней мере, одним значением интервала Tari, которое должно быть в диапазоне от 8 мкс до 25 мкс, и хотя бы одним значением параметра x = 1,5 Tari и x = 2,0 Tari.

6.3.3.3.1.2.5 Радиочастотный сигнал «УСО-радиочастотная метка» (R => T)

Метод ASK: огибающая сигнала «УСО-радиочастотная метка» R => T должна соответствовать рисунку 19 и таблице 32. Напряженность магнитного поля A — максимальная амплитуда огибающей

радиочастотного сигнала RF, измеренной в А/м, интервал Tari определен на рисунке 17. Длительность импульса измерена по уровню 50 %.

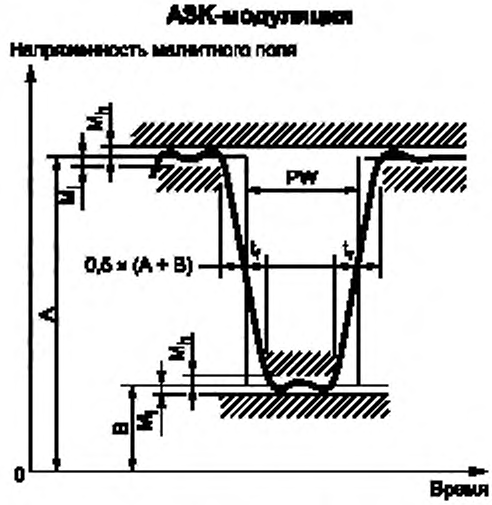



Рисунок 19 — Метод ASK: Огибающая сигнала «УСО-радиочастотная метка»

Таблица 32 — Метод ASK: Параметры огибающей РЧ сигнала

| Интервал Tari, мкс | Параметр | Символ | Минимум | Обычно | Максимум | Единица измерения |
|---|---|---|---|---|---|---|
| От 8 до 25 | Коэффициент модуляции | (A–B)/(A+B) | 10 | 15 | 30 | % |
| | Положительный выброс пульсации РЧ огибающей | Mh | 0 | | 0,1 (A—B) | А/м |
| | Отрицательный выброс пульсации РЧ огибающей | Ml | 0 | | 0,1 (A—B) | А/м |
| От 8 до 25 | Время нарастания РЧ огибающей | $t_{r,10-90\ \%}$ | 0 | | MIN(0,33 Tari, 4,5) | мкс |
| | Время спада РЧ огибающей | $t_{f,10-90\ \%}$ | 0 | | MIN(0,33 Tari, 4,5) | мкс |
| | Ширина импульса РЧ сигнала | PW | MAX (0,265 Tari, 4) | | MIN (0,525 Tari, 9,44) | мкс |

Метод PJM: Связь «УСО-радиочастотная метка» должна использовать метод PJM (фазовую модуляцию колебаний). Данные передаются как очень маленькие изменения фазы несущей частоты УСО. Форма сигнала сдвига фаз магнитного поля УСО методом PJM показана на рисунке 20 и в таблице 33.

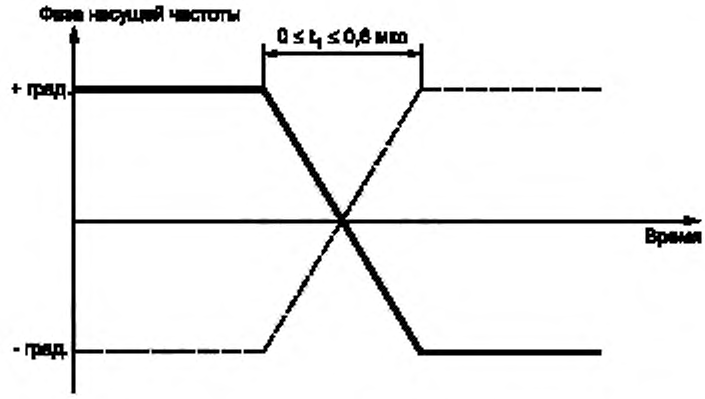



Рисунок 20 — Метод PJM: схема модуляции команд

Таблица 33 — Параметры модуляции команды

| Параметр | Обозначение | Номинальное | Максимальное | Единица измерения |
|---|---|---|---|---|
| Сдвиг фазы | +градус,–градус | 3,0 | 6,0 | градус |
| Время перехода | $t_1$ | 0,0 | 0,6 | мкс |

Примечание — Фазовый сдвиг не может превышать конечное значение фазы в любое время, время перехода ($t_1$) представляет собой время, в течение которого достигается значение 95 % от полного изменения фазы.

6.3.3.3.1.2.6 Форма сигнала включения питания УСО

Огибающая радиочастотного сигнала включения УСО должна соответствовать рисунку 21 и таблице 34. Как только уровень несущей становится выше 10 % уровня, огибающая сигнала включения должна расти монотонно по крайней мере до уровня MI. Огибающая не должна опускаться ниже уровня 90 % (см. рисунок 21) или подниматься выше уровня 110 % во время интервала $T_s$. После интервала $T_s$ огибающая не должна опускаться ниже 99 % или подниматься выше 101 %. УСО не должны передавать команды до окончания интервала максимального времени установления (см. таблица 34 (т. е. до $T_s$)). УСО должны удовлетворять требованиям к точности частоты, указанным в 6.3.3.3.1.2.1, к моменту окончания интервала $T_s$ (см. рисунок 21).

Примечание — При проведении теста на соответствие указанным требованиям в поле не должно быть никаких движущихся меток.

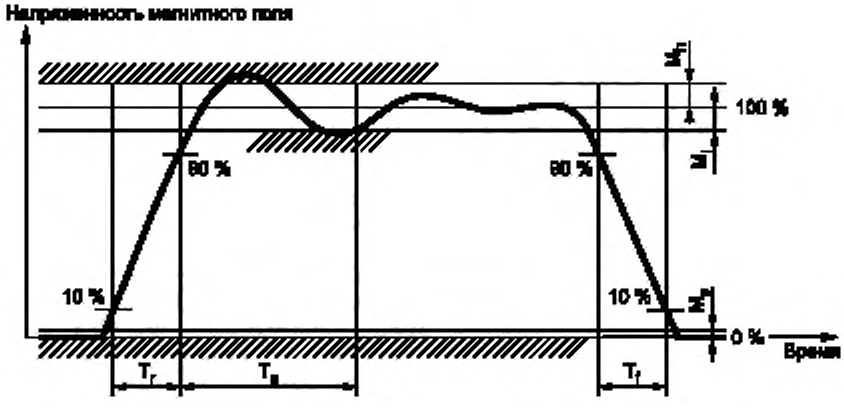



Рисунок 21 — РЧ огибающая включения и выключения питания УСО

6.3.3.3.1.2.7 Форма сигнала выключения питания УСО

РЧ огибающая выключения питания УСО должна соответствовать показанной на рисунке 21 и таблице 34. Как только уровень несущей упадет ниже уровня 99 %, огибающая выключения питания должна монотонно спадать до предела выключения питания $M_s$. Как только выключено питание, УСО должно оставаться выключенным не менее 1 мс перед повторным включением питания.

Таблица 34 — Параметры сигнала включения питания УСО

| Параметр | Определение | Минимум | Обычно | Максимум | Единица измерения |
|---|---|---|---|---|---|
| $T_r$ | Время нарастания | 1 | — | 500 | мкс |
| $T_s$ | Время установления | — | — | 1500 | мкс |
| $M_s$ | Уровень сигнала, когда ВЫКЛ | — | — | 0,1 | % полной шкалы |
| $M_l$ | Отрицательный выброс пульсации | — | — | 10 | % полной шкалы |
| $M_h$ | Выброс пульсации | — | — | 10 | % полной шкалы |

Таблица 35 — Параметры сигнала выключения питания УСО

| Параметр | Определение | Минимум | Обычно | Максимум | Единица измерения |
|---|---|---|---|---|---|
| $T_f$ | Время затухания | 1 | — | 500 | мкс |
| $M_s$ | Уровень сигнала, когда ВЫКЛ | — | — | 0,1 | % полной шкалы |

6.3.3.3.1.2.8 R => T заголовок и синхронизация кадров

Метод ASK: УСО должно начать осуществлять связь R=>T либо с заголовком, либо с синхронизацией кадра, оба варианта показаны на рисунке 22. Заголовок должен предшествовать команде BeginRound (см. 6.3.3.4.11.2.1) и обозначать начало цикла инвентаризации. Любая другая связь должна начинаться с кадра синхронизации. Допуск на все параметры, указанные в единицах интервала Tari, должен быть ±1 %. PW должна быть, как указано в таблице 32. Огибающая РЧ должна быть как указано на рисунке 19. Радиочастотная метка может приравнивать длину 0 данных к длине RTcal для подтверждения правильности заголовка.

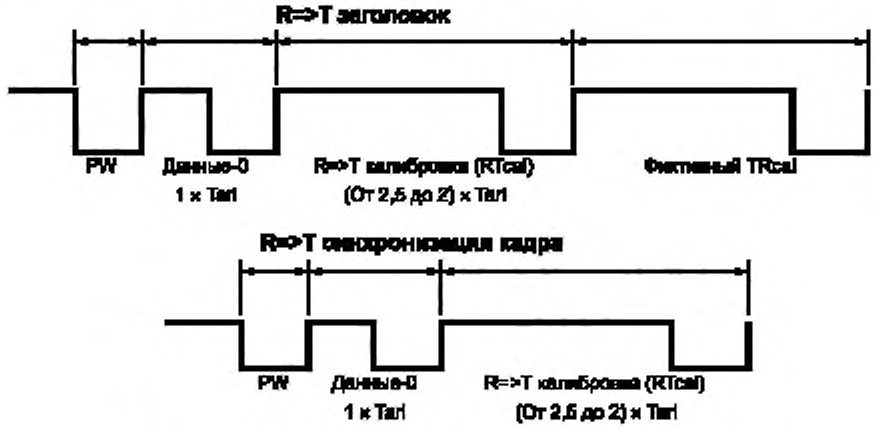



Рисунок 22 — Метод ASK: R => T заголовок и синхронизация кадров

Заголовок должен содержать модуляцию с той же длиной, которая используется в следующем символе **0**, символ **0**, символ калибровки R=>T (RTcal) и фиктивный символ калибровки T=>R (TRcal).

- RTcal: УСО должно устанавливать RTcal равным длине символа 0 плюс длина символа данные-1 (RTcal = 0-длина + 1-длина). Радиочастотная метка должна измерять длину RTcal и вычислять опорную точку = RTcal/2. Радиочастотная метка должна интерпретировать последующие символы УСО короче

опорной точки как 0, а последующие символы УСО длиннее опорной точки как 1. Радиочастотная метка должна интерпретировать символы длиннее, чем 4 RTcal как недействительные. Перед изменением RTcal УСО должно передать CW длительностью как минимум 8 RTcal.

- TRcal: В высокочастотном протоколе для определения T=>R соотношения обратного соединения TRcal не используется. Величина TRcal должна быть в диапазоне от 1,1*RTcal до 3*RTcal (1.1*RTcal<=TRcal<=3*RTcal).

- Кадр синхронизации тождественен заголовку минус символ TRcal. Для длительного цикла инвентаризации УСО должно в кадре синхронизации использовать ту же длину RTcal, которая использовалась в заголовке, начавшем цикл.

Метод PJM: в соединении R => T должна использоваться PJM с MFM кодированием. УСО должно начинать все R=>T связи с флага MFM (или кадром синхронизации). Флаг определяет начало команды и синхронизацию битового интервала. Флаг состоит из трех частей:

- Синхронизирующая последовательность 9 бит достоверных MFM закодированных данных. В качестве примера на рисунке 23 для двух возможных флагов команд показаны 9 бит данных со значением «1».

- В обычных данных нарушения MFM кодирования нет. Нарушение представляет собой последовательность из 4 изменений состояния, разделенных 2-битовым интервалом, 1,5-битовом интервалом, и 2-битовым интервалом. Граница четвертого перехода определяет начало битового интервала.

- Нулевой байт в конце строки определяет конец флага.

Синхронизирующая строка, нарушения кодирования и конечный ноль для двух возможных флагов команд показаны на рисунке 23.

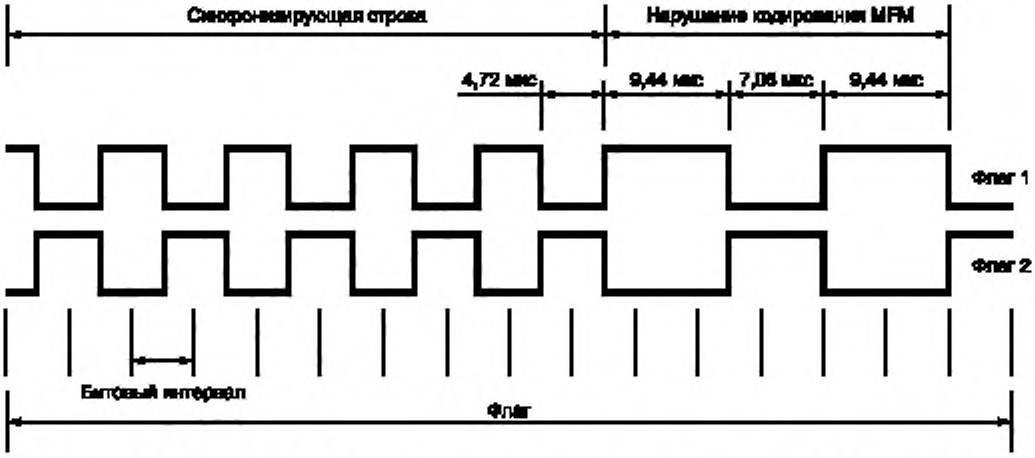



Рисунок 23 — Метод PJM: MFM кодирование и синхронизация для двух возможных флагов команд

6.3.3.3.1.3 Связь «радиочастотная метка-УСО» (T => R)

Для связи радиочастотной метки с УСО используется модуляция нагрузкой. Модуляция нагрузкой — это метод наложения цифрового сигнала на несущую частоту путем изменения электрической «нагрузки» (импеданса) радиочастотной метки. Радиочастотная метка способна передавать данные путем изменения ее «нагрузки», так осуществляется изменение напряжения в антенне УСО. УСО может передавать изменения напряжения в двоичном сигнале.

Модуляция нагрузкой может быть достигнута несколькими путями, например переключением:

- сопротивления нагрузки;
- шунтирующего диода;
- перестройкой значения емкости антенны или отвода на антенной катушке (регулировка свободного конца). Можно рассматривать переключаемую настроечную емкость как часть антенны, и поэтому нагрузка в течение ответа воспринимается антенной радиочастотной метки как фиксированная. Аналогичным образом настроечный конденсатор и нагрузка чипа для отвода катушки считаются постоянными.

Метод ASK: УСО должно выполнять модуляцию нагрузкой, используя фиксированные формат модуляции, кодирование данных и скорость передачи данных в битах в течение всего цикла инвентаризации, где «цикл инвентаризации» определен в 6.3.3.4.8. УСО задает кодирование и скорость передачи данных в битах с помощью команды *BeginRound*, которая запускает цикл. По команде *BeginRound* радиочастотная метка выбирает вид модуляции T=>R.

Метод PJM: скорость передачи данных в битах ответа — 106 Кбит/с, кодирование с использованием MFM, а модуляция на поднесущей — BPSK. Радиочастотные метки могут выбрать одну из восьми поднесущих частот между 969 кГц и 3013 кГц. Поднесущая является производной от деления рабочей частоты УСО. Во время цикла инвентаризации радиочастотная метка должна выполнять модуляцию нагрузкой, используя постоянную частоту поднесущей, как описано в подразделе метода PJM 6.3.3.1.3.10. УСО задает канал ответа с помощью команды *BeginRound*, которая запускает цикл.

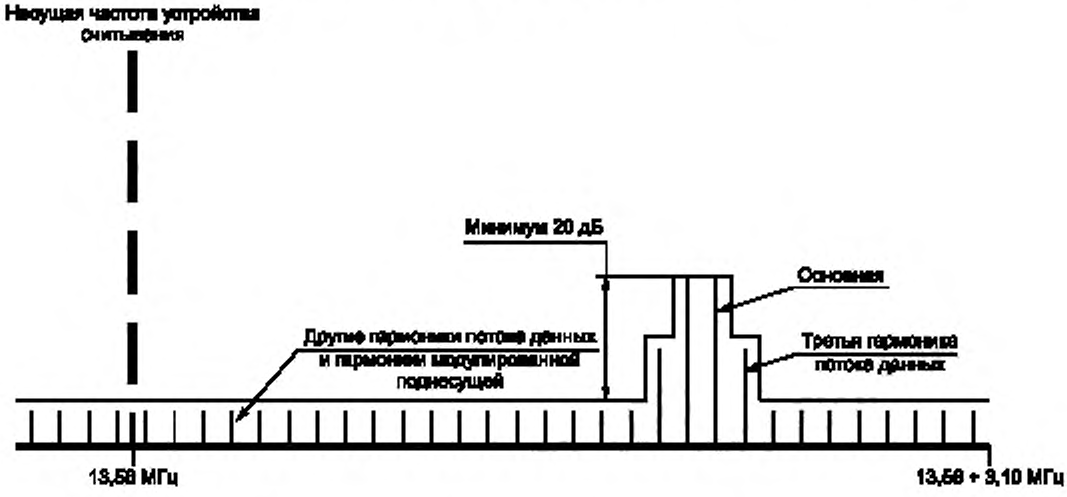
6.3.3.3.1.3.1 Модуляция

Для последующих рисунков нижние значения соответствуют антенне радиочастотной метки, имеющей импеданс нагрузки равный импедансу нагрузки во время интервала CW непосредственно перед передачей данных, тогда как верхние значения соответствуют антенне радиочастотной метки, имеющей другой импеданс нагрузки для интервала CW.

- Метод ASK: Сигнал ответа радиочастотной метки может быть полосовым (для FM0 кодирования) или модулированной поднесущей (для кода Миллера или Манчестерского кодирования), как указано в пунктах от 6.3.3.3.1.3.2 до 6.3.3.3.1.3.10.

- Метод PJM: сигнал ответа радиочастотной метки представляет собой модулированную поднесущую BPSK.

- Чтобы обеспечить ответ радиочастотной метки на различных каналах одновременно с получением ее ответа, полоса частот ограничивается, чтобы сократить данные и уменьшить уровни гармоник поднесущей частоты, как показано на рисунке 24. Промежуточные значения импеданса модуляции нагрузкой могут использоваться для целей ограничения полосы.




Примечания

1 Это значение показывает верхнюю боковую полосу частот ответа радиочастотной метки.

2 Так же маска должна быть применена и к нижней боковой полосе.

Рисунок 24 — Метод PJM: Маска ответа радиочастотной метки

6.3.3.3.1.3.2 Кодирование данных

Метод ASK: Данные радиочастотной метки должны кодироваться модуляцией нагрузкой в ответе как полоса FM0, либо Манчестерским кодированием поднесущей, либо с помощью кода Миллера. УСО задает вариант кодирования и скорость передачи данных в битах с помощью параметров в команде *BeginRound*.

Метод PJM: Данные радиочастотной метки должны кодироваться модуляцией нагрузкой в ответе как MFM одной из восьми поднесущих разных частот на скорости передачи данных 106 Кбит/с. Выбор частоты поднесущей производится либо случайным выбором радиочастотной метки, либо прямым указанием в параметре команды *BeginRound*.

6.3.3.3.1.3.3 Метод ASK: FM0 полосы частот

На рисунке 25 показаны базовые операции для реализации FM0 кодирования данных (двухфазное пространство). FM0 инвертирует фазу полосы частот на границе каждого символа; **0** данные имеют дополнительную инверсию фазы среднего символа.

Рисунок 26 показывает генерацию FM0 полосы символов и последовательностей. Рабочий цикл последовательности **00** или **11**, измеренный на выходе модулятора, должен быть 50 %. FM0 кодирование обладает памятью; поэтому выбор последовательности FM0 на рисунке 26 зависит от состояния после предыдущей передачи. Связь FM0 должна всегда оканчиваться специальным символом EOF в конце передачи, как показано на рисунке 28.

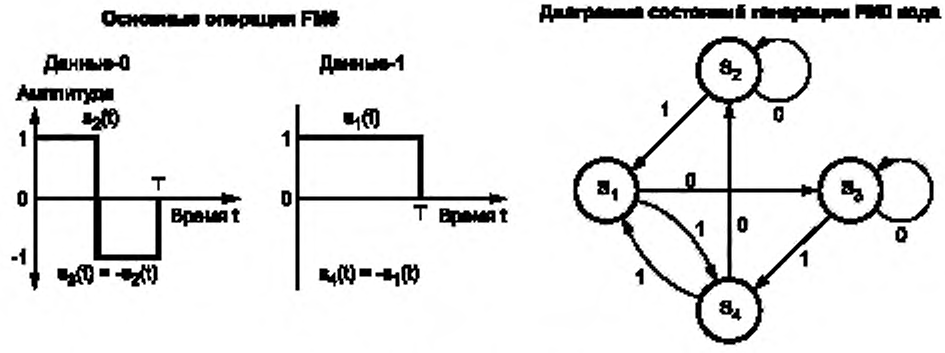



Рисунок 25 — Метод ASK: операции FM0
и диаграмма состояний генерации кода

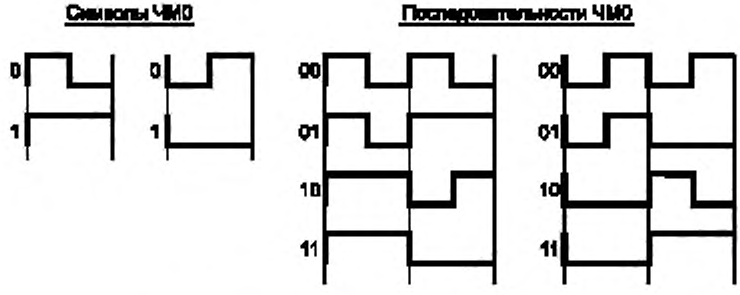



Рисунок 26 — Метод ASK: FM0 символы
и последовательности

6.3.3.3.1.3.4 Метод ASK: FM0 заголовок SOF и символ EOF

Связь FM0 T=>R должна начинаться с одного из двух заголовков, показанных на рисунке 27. Заголовок содержит четыре **0** данных, за которыми следует нарушение кодирования FM0 длительности одного бита, а оканчивается одним **0** данных. Выбор в зависимости от значения бита TRext указан в команде *BeginRound*, которая запускает цикл инвентаризации. На рисунке 27 пробный сигнал для TRext=1 состоит из дополнительных непрерывных периодов на частоте поднесущей на протяжении 12 бит. Все ответы радиочастотной метки заканчиваются символом EOF, показанным на рисунке 28.

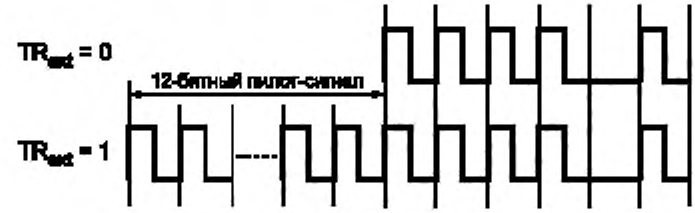



Рисунок 27 — Метод ASK: FM0 заголовок SOF

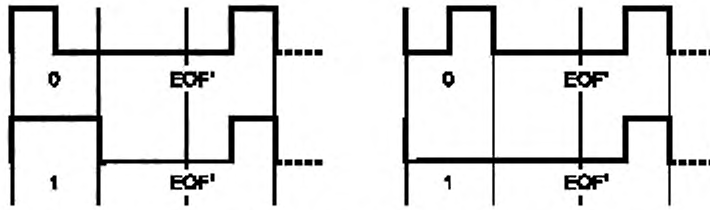



Рисунок 28 — Метод ASK: символ EOF, завершающий FM0 передачу

6.3.3.3.1.3.5 Метод ASK: Манчестерское кодирование поднесущей

На рисунке 29 показаны основные операции для реализации Манчестерского кодирования. Манчестерское кодирование использует переключение с высокого на низкий уровень для **0** и переключение с низкого на высокий уровень для **1**.

Модуляция поднесущей для Манчестерского кодирования **0** предусматривает импульсы поднесущей, за которым следует немодулированный интервал. Данные **1** кодируются немодулированным интервалом, за которым следуют импульсы поднесущей.

На рисунке 30 показана Манчестерское кодирование последовательности на поднесущей; Манчестерское кодирование последовательности должно содержать точно четыре или восемь периодов на один бит данных, в зависимости от значения M, указанного в команде *BeginRound*, запускающей цикл инвентаризации (см. таблица 36). Два возможных значения частоты поднесущей (fc/32 и fc/16) также определяются в команде *BeginRound* битом DR.

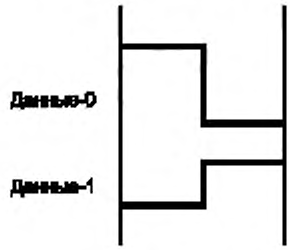



Рисунок 29 — Метод ASK: Основные операции Манчестерского кодирования

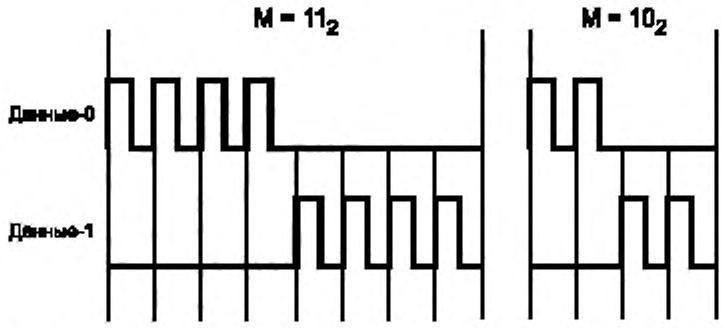



Рисунок 30 — Метод ASK: Манчестерское кодирование последовательности на поднесущей

6.3.3.3.1.3.6 Метод ASK: Заголовок Манчестерского кодирования поднесущей SOF и символ EOF

Связь поднесущей T=>R должна начинаться с одного из двух заголовков, показанных на рисунке 31. Они содержат точно шесть (или двенадцать) импульсов поднесущей, за которыми следует немодулированный интервал из четырех (или восьми) периодов поднесущей (что соответствует нарушению Манчестерского кодирования) и заканчивается битом «**0**». Выбор в зависимости от значения бита TRext указан в команде *BeginRound*, которая запускает цикл инвентаризации. Продолжительность пробного сигнала для TRext=1 составляет 12 бит. Этот пробный сигнал состоит из непрерывных периодов на частоте поднесущей. Все ответы радиочастотной метки заканчиваются символом EOF, показанным на рисунке 32. Символ EOF содержит бит «**1**», за которым следует нарушение Манчестерского кодирования в виде группы из четырех (или восьми) импульсов поднесущей.

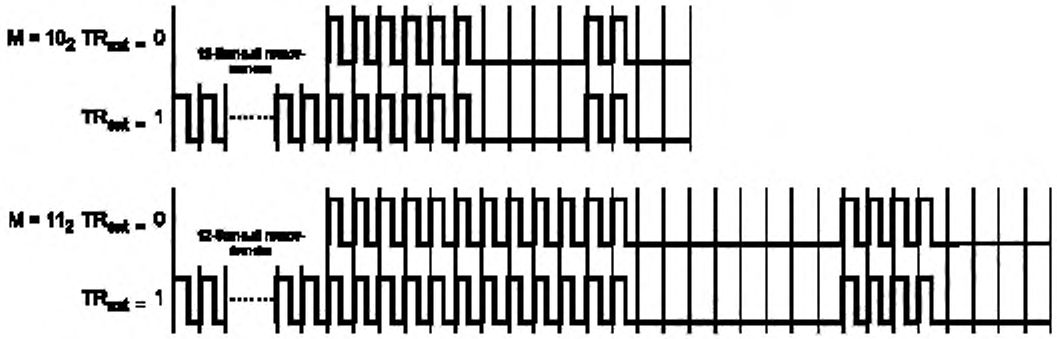



Рисунок 31 — Метод ASK: Поднесущая T=>R, заголовок SOF

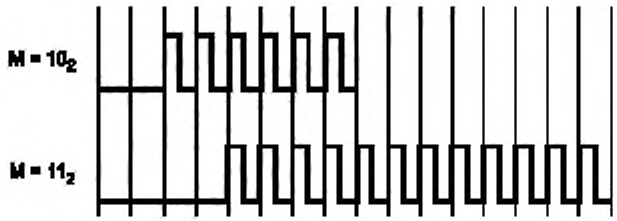



Рисунок 32 — Метод ASK: Символ EOF завершающий передачу поднесущей

6.3.3.3.1.3.7 Модуляция кодом Миллера поднесущей

На рисунке 33 показаны основные операции для реализации кода Миллера. Сигнал основной полосы в коде Миллера инвертирует свою фазу между двумя **0** данных, стоящими рядом в последовательности. Основная полоса сигнала в коде Миллера также инвертирует фазу в середине **1** данных. Диаграмма состояний, показанная на рисунке 33, отображает логическую последовательность состояний при реализации основных операций кода Миллера над данными. Узловые элементы диаграммы S1—S4 показывают четыре возможных значения данных, которые можно закодировать двумя фазами базовых операций кода Миллера. Значения на дугах диаграммы представляют собой код Миллера, который генерируется при переходе в конечное состояние. Форма передаваемого сигнала соответствует основной полосе сигнала прямоугольной формы с увеличенным в восемь раз временем передачи символа данных. Значения, указанные на смене состояний, обозначают логические значения последовательности данных при кодировании. Например, переход из состояния S1 в S3 не разрешен, так как в результате перехода произойдет инверсия фазы на границе символа между данными **0** и данными **1**.

На рисунке 34 показана последовательность кода Миллера на поднесущей; последовательность кода Миллера должна содержать ровно восемь периодов поднесущей на бит (см. таблица 36). Кодирование кодом Миллера обладает памятью; поэтому выбор последовательности поднесущей Миллера, показанной на рисунке 34, зависит от состояния после предыдущей передачи.

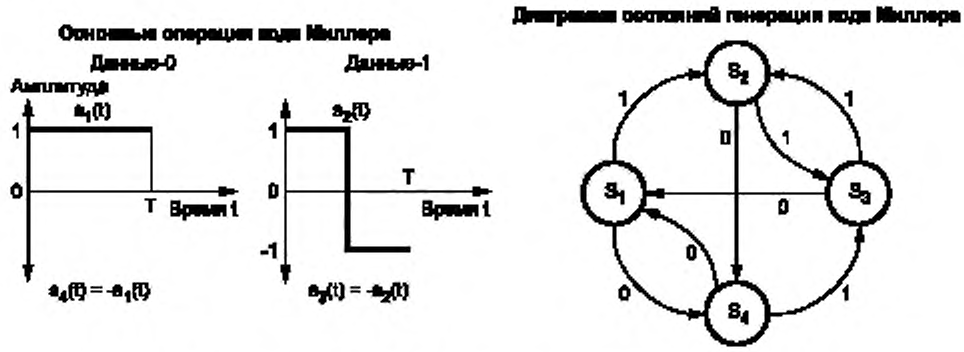



Рисунок 33 — Метод ASK: Основные операции кода Миллера
и диаграмма состояний генерации кода

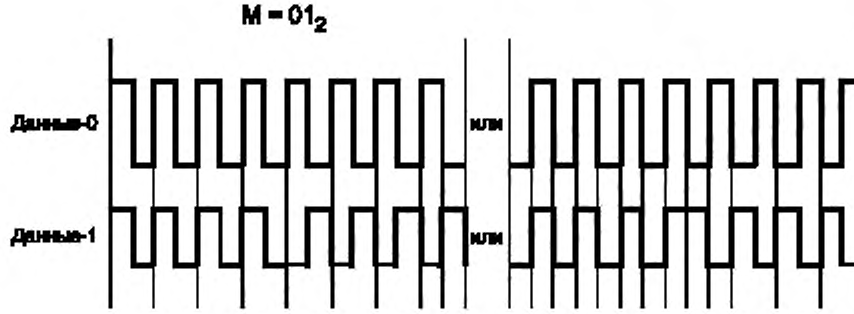



Рисунок 34 — Метод ASK: Последовательность кода Миллера на поднесущей

6.3.3.3.1.3.8 Метод ASK: Заголовок SOF и символ EOF поднесущей с кодом Миллера

Связь T=>R на поднесущей должна начинаться с одного из двух возможных заголовков, показанных на рисунке 35. Выбор зависит от значения бита TRext, указанного в команде *BeginRound*, которая запускает цикл инвентаризации. На рисунке 35 показано время пилотного тонового сигнала для каждого TRext. Пилотный сигнал состоит из непрерывных периодов на частоте поднесущей. Обмен данными посредством кода Миллера должен всегда оканчиваться специальным символом EOF в конце передачи, как показано на рисунке 36.

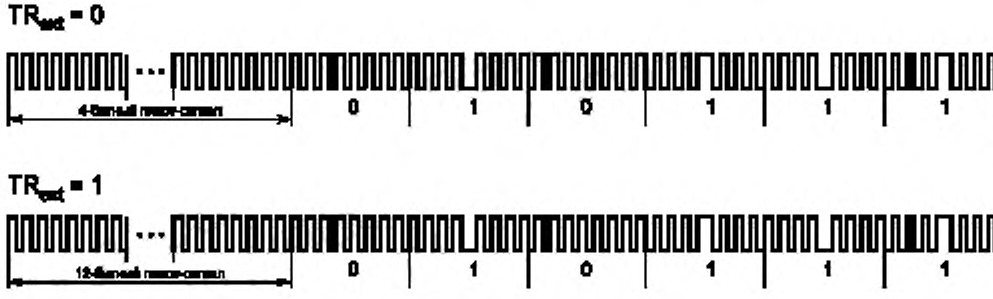



Рисунок 35 — Метод ASK: Заголовок SOF поднесущей
с кодом Миллера для связи T=>R

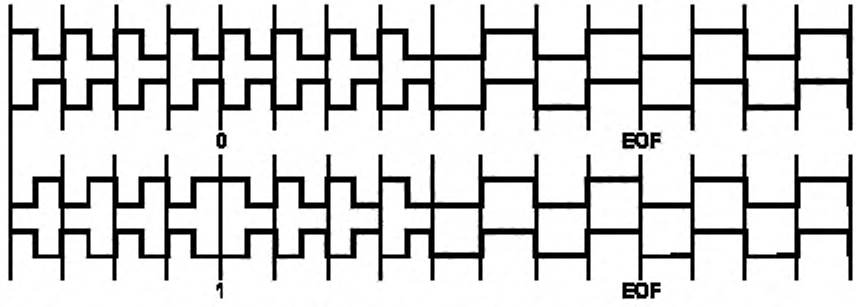



Рисунок 36 — Метод ASK: Символ EOF, завершающий передачу поднесущей

6.3.3.3.1.3.9 Метод PJM: MFM модуляция поднесущей

На рисунке 37 показаны основные операции MFM кодирования. Сигнал основной полосы MFM инвертирует свою фазу между двумя **0** в последовательности данных. Сигнал основной полосы MFM также располагает инверсию фазы в середине символа **1** данных. Диаграмма состояний на рисунке 37 отображает логическую последовательность состояний при реализации основных операций MFM кодирования на сигнале основной полосы. Узловые элементы диаграммы S1—S4 показывают четыре возможных значения данных MFM-кодирования, представленных двумя фазами каждой базовой операции MFM. Эти значения также представляют собой форму сигнала основной полосы MFM, который генерируется на основании начального состояния. Форма передаваемого сигнала — это форма сигнала основной полосы, умноженная на прямоугольный сигнал частоты $f_c$ деленной на $n$, где $n$ — коэффициент деления для одной из 8 поднесущих, определенный в таблице 38. Значения на дугах диаграммы при смене состояний обозначают логическую связь значений данных в кодируемой последовательности. Например, переход из состояния $S_1$ в $S_3$ не разрешен, так как в результате перехода произойдет инверсия фазы на границе символа между данными **0** и данными **1**.

На рисунке 38 показан пример изменения фазы двоичной фазовой манипуляции (BPSK) вне поднесущей и соответствующая ей последовательность MFM-модулирования поднесущей. Фазовые переходы выравниваются при изменениях состояния MFM, как показано на рисунке 37. Для наглядности в качестве примера показаны восемь периодов поднесущей на битовый интервал, однако в действительности будет от 9 до 28 периодов поднесущей в зависимости от выбранного канала приема. Также для наглядности показано, что изменение фазы происходит в начале и в середине периода поднесущей; однако изменение фазы может происходить в любой точке внутри периода поднесущей, так как скорость передачи данных в битах ответа и поднесущие не сихронизированы.

Поднесущая получается делением частоты силового питающего поля и определяется из значений DR, M и TRext, указанных в команде *BeginRound*, которая запускает цикл инвентаризации (см. таблица 38). Период битового интервала, используемого для кодирования командной информации, составляет 9,44 мкс. MFM-кодирование обладает памятью; поэтому выбор последовательности MFM на рисунке 38 зависит от состояния после предыдущей передачи.

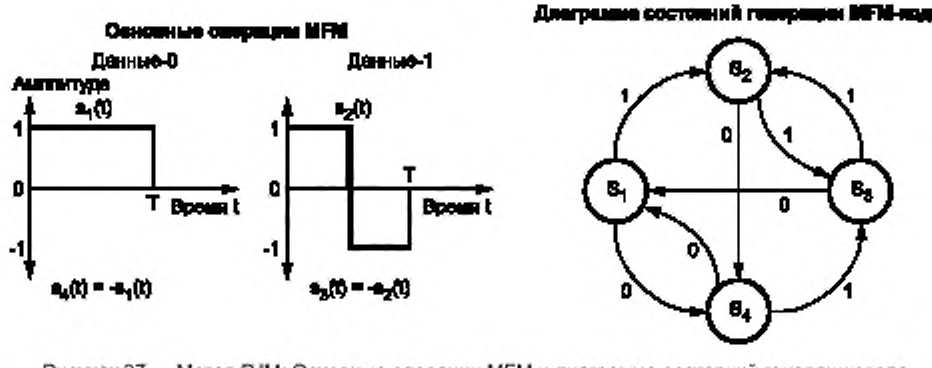



Рисунок 37 — Метод PJM: Основные операции MFM и диаграмма состояний генерации кода

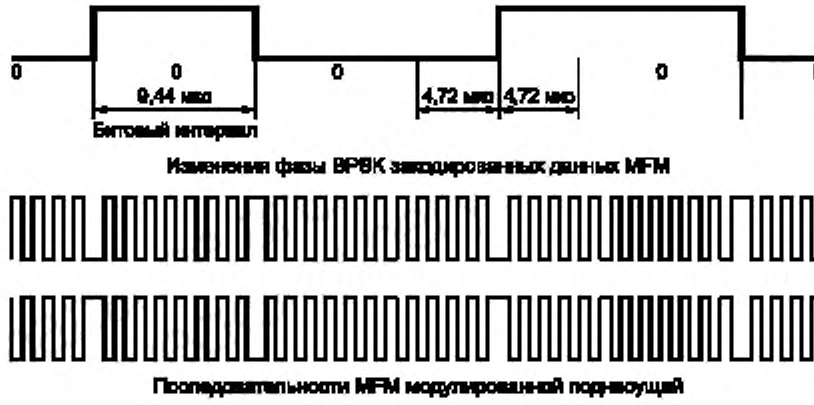



Рисунок 38 — Метод PJM: MFM кодирование ответа
и временная диаграмма двоичной последовательности 000100

6.3.3.3.1.3.10 Метод PJM: Флаг MFM поднесущей (заголовок SOF)

Метод PJM: Связь T=>R на поднесущей должна начинаться с любого из двух заголовков, показанных на рисунке 39. Этот рисунок демонстрирует синхронизацию флага при изменениях фазы поднесущей двоичной фазовой манипуляции (BPSK).

Флаг определяет начало ответа и распределение по времени битовых интервалов. Флаг состоит из трех частей:

- синхронизирующая последовательность, состоящая из 9 бит допустимых данных, закодированных MFM. В качестве примера на рисунке 39 показаны 9 бит данных со значением "1";
- нарушение кодирования MFM, которое представлено в обычной передаче данных. Нарушение представляет собой последовательность из 4 изменений состояния, разделенных 2-битовым интервалом, 1,5-битовым интервалом и 2-битовым интервалом. Граница четвертого перехода определяет начало битового интервала;
- завершающий бит 0 в конце строки.

Синхронизирующая строка, нарушение кодирования и конечный ноль для двух возможных флагов ответа показаны на рисунке 39.

Примечание — Символ EOF не требуется.

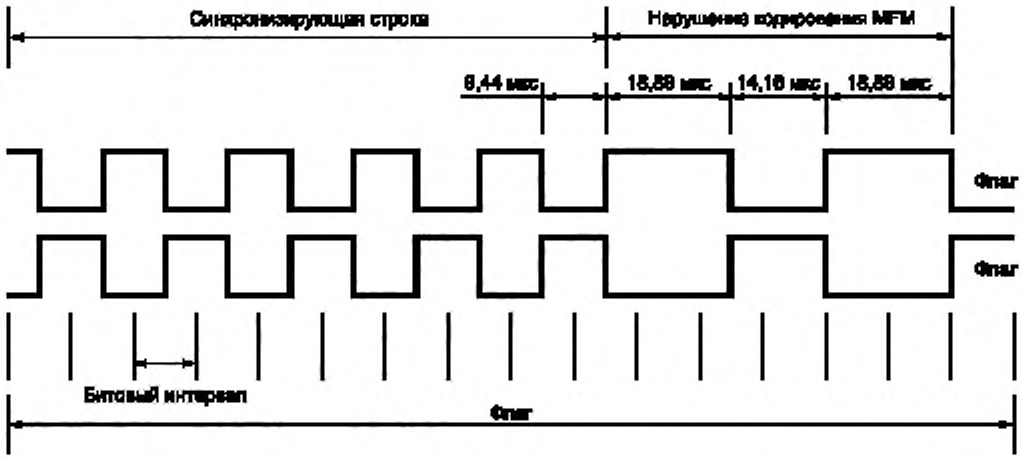



Рисунок 39 — Метод PJM: Флаги поднесущей T=>R

6.3.3.3.1.3.11 Метод ASK и Метод PJM: Выбор вида модуляции и скорости передачи данных

- Метод ASK: Радиочастотные метки должны поддерживать значения R=>T Tari, указанные в 6.3.3.3.1.2.4, T=>R частоты соединения, указанные в таблице 36, и T=>R скорости передачи данных, указанные в таблице 37. Команда *BeginRound*, которая запускает цикл инвентаризации, вычисляет DR в таблице 36 и M в таблице 37.

Таблица 36 — Метод ASK: Частоты соединения канала «радиочастотная метка-УСО»

| Значения M | Бит модуляции | Поднесущая с DR=0 LF = 423 кГц ($f_c$/32), Кбит/с | Поднесущая с DR=1 LF = 847 кГц ($f_c$/16), Кбит/с |
|---|---|---|---|
| $11_2$ | 4 импульса поднесущей Манчестерского кодирования | 53 ($f_c$/256) | 106 ($f_c$/128) |
| $10_2$ | 2 импульса поднесущей Манчестерского кодирования | 106 ($f_c$/128) | 212 ($f_c$/64) |
| $01_2$ | 8 импульсов поднесущей закодированной кодом Миллера | 53 ($f_c$/256) | 106 ($f_c$/128) |
| $00_2$ | ЧМ0 | 424 ($f_c$/32) | 848 ($f_c$/16) |

Таблица 37 — Метод ASK: Соединение «радиочастотная метка-УСО», скорость передачи битовых данных

| M: Число периодов поднесущей на символ | Вид модуляции | Скорость передачи битовых данных, Кбит/с |
|---|---|---|
| M = $00_2$: 1 период на символ | ЧМ0 полосы частот | LF |
| M = $01_2$: 8 периодов на символ | Поднесущая с кодом Миллера | LF/8 |
| M = $10_2$: 4 периода на символ | Поднесущая с Манчестерским кодированием | LF/4 |
| M = $11_2$: 8 периодов на символ | Поднесущая с Манчестерским кодированием | LF/8 |

- Метод PJM: T=>R должна быть MFM модуляция при скорости передачи данных 106 Кбит/с на поднесущей, выбранной командой BeginRound, которая запускает цикл инвентаризации. Команда *BeginRound* содержит биты DR, M и TRext, которые (для команд метода PJM) декодируются радиочастотными метками, поддерживающими метод, как показано в таблице 38.

Таблица 38 — Метод PJM: Команды выбора поднесущей

| DR | M | TRext | Выбор канала ответа | Коэффициент деления | Поднесущая частота, кГц |
|---|---|---|---|---|---|
| 0 | 00 | 0 | A | 14 | 969 |
| 0 | 00 | 1 | B | 11 | 1233 |
| 0 | 01 | 0 | C | 9 | 1507 |
| 0 | 01 | 1 | D | 7,5 | 1808 |
| 0 | 10 | 0 | E | 6,5 | 2086 |
| 0 | 10 | 1 | F | 5,5 | 2465 |
| 0 | 11 | 0 | G | 5 | 2712 |
| 0 | 11 | 1 | H | 4,5 | 3013 |
| 1 | 00 | 0 | Скачкообразная перестройка частоты поднесущей | | |
| Все другие состояния | | | Зарезервировано для будущего использования | | |

Если командой *BeginRound* указан режим скачкообразной перестройки частоты поднесущей*, радиочастотная метка должна выбрать канал ответа. Радиочастотная метка должна использовать выбранный канал ответа для всех T=>R соединений. Канал, выбранный радиочастотной меткой, должен быть определен из трех самых младших значащих битов кода StoredCRC. Три младших значащих бита StoredCRC должны использоваться как значения битов M и TRext в таблице 38, приведенной выше. Например, если три младших значащих бита кода Stored CRC имеют значение **110**, то должен быть выбран канал G.

6.3.3.3.1.4 Оба метода: порядок передачи

Для всех R=>T и T=>R соединений должны соблюдаться следующие правила:

- В каждом сообщении самое старшее слово должно передаваться первым;
- В каждом сообщении старший значащий бит должен передаваться первым.

6.3.3.3.1.5 Оба метода: контроль циклическим избыточным кодом (CRC)

CRC — это циклический избыточный код, который используется радиочастотной меткой, чтобы гарантировать достоверность принятых команд при соединении R=>T, а УСО использует его для того, чтобы гарантировать достоверность полученных ответов при соединении T=>R. Этот протокол контроля использует два типа CRC: (i) CRC-16, и (ii) CRC-5. В Приложении I описаны оба типа CRC.

Для генерации CRC-16 радиочастотная метка или УСО должны сначала сгенерировать значение прекурсора CRC-16, как показано в таблице 39, а затем получить его дополнение** для формирования значения CRC-16.

Таблица 39 — Прекурсор CRC 16

| Тип CRC | Длина | Полином | Предустановка | Вычет |
|---|---|---|---|---|
| *[7]* | 16 бит | $x^{16} + x^{12} + x^5 + 1$ | $FFFF_h$ | $1D0F_h$ |

Радиочастотная метка или УСО должны проверять достоверность принятого сообщения, используя код CRC-16. Для реализации проверки CRC-16 радиочастотная метка или УСО могут использовать один из методов, описанных в Приложении I.

При включении питания радиочастотная метка рассчитывает и записывает в память 16-разрядный код Stored CRC — см. 6.3.3.4.1.2.1.

Радиочастотные метки должны добавлять код CRC-16 к тем ответам, которые используют CRC-16 (см. 6.3.3.4.11) как это предусмотрено форматом команд.

Для генерации CRC-5 УСО должно использовать описание, приведенное в таблице 40 (см. также приложение I).

Таблица 40 — Описание CRC-5

| Тип CRC | Длина | Полином | Предустановка | Вычет |
|---|---|---|---|---|
| — | 5 бит | $x^5 + x^3 + 1$ | $01001_2$ | $00000_2$ |

Радиочастотная метка должна проверить достоверность принятого сообщения, используя код CRC-5. Для реализации проверки CRC-5 РЧ метка может использовать метод, описанный в Приложении I.

УСО должно добавлять соответствующий CRC к R=>T передаче, как показано в таблице 46.

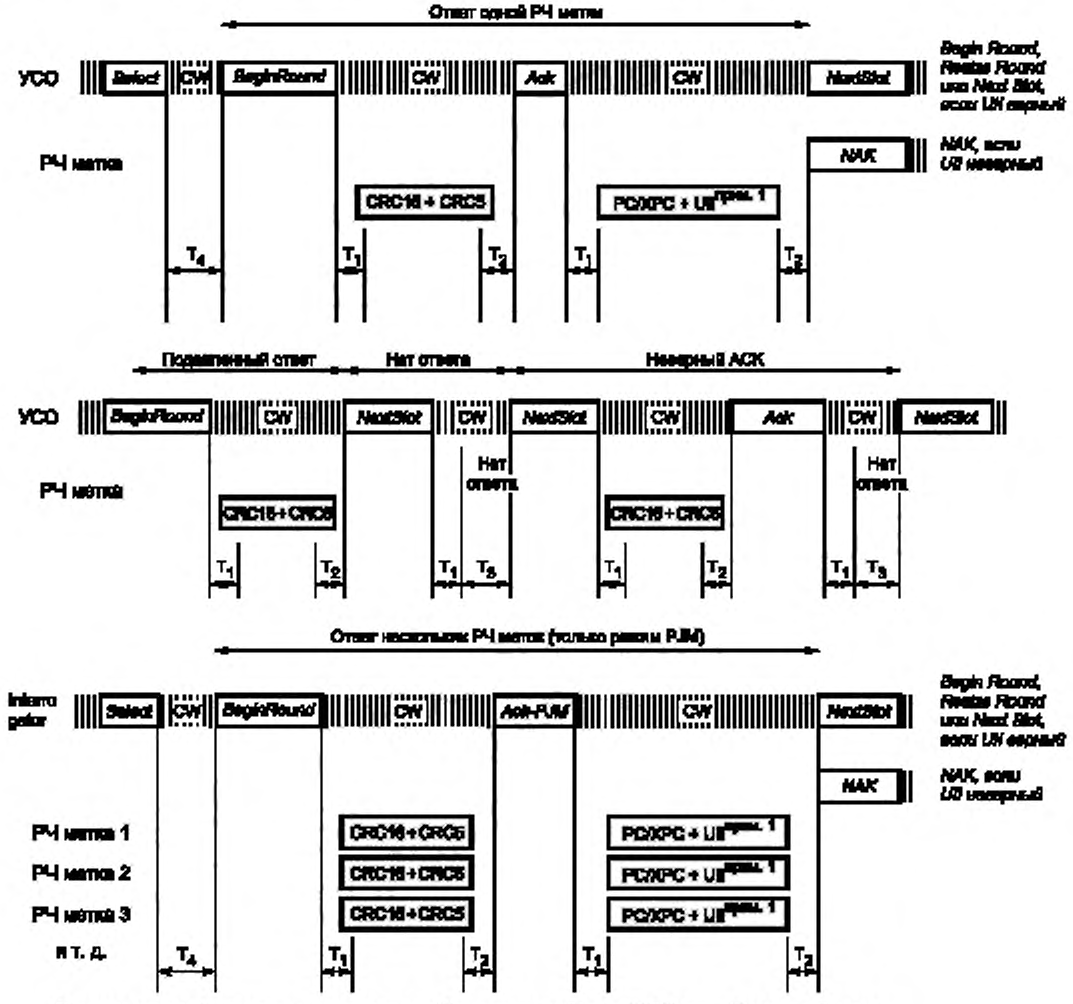
6.3.3.3.1.6 Синхронизация соединения — оба метода

Рисунок 40 иллюстрирует синхронизацию соединений R=>T и T=>R. Рисунок 40 (не в масштабе) определяет взаимодействие УСО с множеством радиочастотных меток. В таблице 41, показаны требуемые параметры синхронизации для рисунка 40, а в 6.3.3.4.11 дано описание команд. Радиочастотные метки и УСО должны соответствовать всем требованиям синхронизации, показанным в таблице 41. RTcal описан в 6.3.3.3.1.2.8. Как изложено в 6.3.3.3.1.2.8, УСО должно использовать фиксированную

* *Алгоритм псевдослучайной перестройки рабочей частоты [FHSS].*

** *Дополнительный двоичный код.*

скорость соединения R=>T на протяжении всего цикла инвентаризации; перед изменением скорости соединения R=>T УСО должно передать CW*, состоящий минимум из 8 RTcal**.




**Примечание 1 – В некоторых случаях добавляется код PackedCRC, если был затребован. Описание возможных случаев см. в таблице 42.**

Рисунок 40 — Временные диаграммы синхронизации соединения — оба метода

Таблица 41 — Параметры синхронизации соединения

| Параметр | Минимум, мкс | Обычно, мкс | Максимум, мкс | Описание |
|---|---|---|---|---|
| $T_1$ | 73,1<br>$(1024-32)/f_c$ | 75,5<br>$(1024/f_c)$ | 77,9<br>$(1024+32)/f_c$ | Время от передачи УСО до ответа радиочастотной метки (особенно время от последнего нарастающего фронта последнего бита, переданного УСО до первого нарастающего фронта ответа радиочастотной метки), измеренного на вводе антенны радиочастотной метки |

* *Непрерывный радиочастотный сигнал. См. 5.2.*

** *Калибровочный символ при передаче R=>T. См. 5.2.*

*Окончание таблицы 41*

| Параметр | Минимум, мкс | Обычно, мкс | Максимум, мкс | Описание |
|---|---|---|---|---|
| $T_2$ | 151 ($2048/f_c$) | — | 1208 ($16384/f_c$) | Время отклика УСО, требующееся, если радиочастотная метка демодулирует сигнал УСО, и измеренное от конца ответа радиочастотной метки до первого спадающего фронта передачи УСО |
| $T_3$ | Tsof_tag | — | — | Время ожидания УСО после $T_1$, перед подачей им другой команды |
| $T_4$ | $T_{1Typ} + T_{3min}$ | — | — | Минимальное время между командами системы опроса |

Примечания

1 Радиочастотная метка может увеличивать максимальное значение $T_1$ при ответе на команду записи в память.

2 Максимальное значение для $T_2$ должно применяться только для радиочастотных меток в состояниях «**Ответ**» или «**Подтверждение**» (см. 6.3.3.4.4.3 и 6.3.3.4.4.4). Для радиочастотных меток в состояниях «**Ответ**» или «**Подтверждение**», если $T_2$ окончилось (т. е. достигнуто его максимальное значение):

- без получения радиочастотной меткой допустимой команды радиочастотная метка должна переключаться в состояние «**Арбитраж**» (см. п. 6.3.3.4.4.2);
- при получении допустимой команды радиочастотная метка должна выполнить команду;
- при получении неверной команды радиочастотная метка должна переходить в состояние «**Арбитраж**» на основании определения того, что команда неверная;
- во всех других состояниях максимальное значение для $T_2$ может быть неограниченным. Понятие «Неверная команда» определено в п. 6.3.3.4.11.

3 При определении того, превышено ли $T_2$, радиочастотная метка должна использовать значение допустимого отклонения $16384/f_c <= T_{2max} <= 16448/f_c$.

6.3.3.4 Выбор, инвентаризация и доступ к радиочастотной метке

Выбор, инвентаризацию и доступ к радиочастотной метке можно рассматривать как низший уровень для уровней передачи данных в многоуровневой модели системы сетевых коммуникаций.

6.3.3.4.1 Память радиочастотной метки

Память радиочастотной метки должна быть логически разделена на четыре отдельных банка, каждый из которых может содержать ноль и более запоминаемых слов. Логическая модель распределения памяти показана на рисунке 41.

Банки памяти — это:

- **Резервная память** должна содержать пароли уничтожения и/или доступа, если пароли установлены на радиочастотной метке. Пароль уничтожения должен храниться в памяти по адресам от $00_h$ до $1F_h$. Пароль доступа должен храниться в памяти по адресам от $20_h$ до $3F_h$. См. 6.3.3.4.1.1.

- **Память уникального идентификатора предмета учета (*UII*)** должна содержать код CRC-16 по адресам $00_h$ до $0F_h$, слово Управления протоколом (PC) по адресам $10_h$ до $1F_h$, код (такой как *UII*, и далее относящийся к *UII*), идентифицирующий объект, к которому радиочастотная метка прикреплена (или должна быть прикреплена), по адресу $20_h$, а если радиочастотная метка выполняет Расширенное управление протоколом (XPC), то либо одно (обязательно) либо два (опционально) слова Расширенного управления протоколом (XPC) начинаются по адресу $210_h$. См. 6.3.3.4.1.2.

- **Память TID** должна содержать 8 бит идентификатора класса распределения по *ГОСТ Р ИСО/МЭК 15963*, размещенных в памяти по адресам от $00_h$ до $07_h$. Память TID должна содержать достаточную идентификационную информацию по адресам выше $07_h$ для однозначной идентификации УСО команд пользователя и/или дополнительных возможностей, которые поддерживает радиочастотная метка. См. 6.3.3.4.1.3.

- **Пользовательская память** является опциональной. См. 6.3.3.4.1.4.

Логическая адресация всех банков памяти должна начинаться с нуля ($00_h$). Физическая схема распределения памяти определяется поставщиком. Команды для доступа в память имеют параметр MemBank, который выбирает банк памяти, параметр длины адреса и параметр адреса для выбора конкретного места расположение в памяти данного банка. Параметр адреса может иметь длину 8, 16, 24 или 32 бита, что определяется в параметре длины адреса. УСО должно использовать самую короткую из возможных длину адреса для кодирования месторасположения памяти. Когда радиочастотная метка

передает содержимое памяти методом модуляции нагрузкой, эта модуляция должна перекрывать собой границы слов (за исключением случая усеченного ответа).

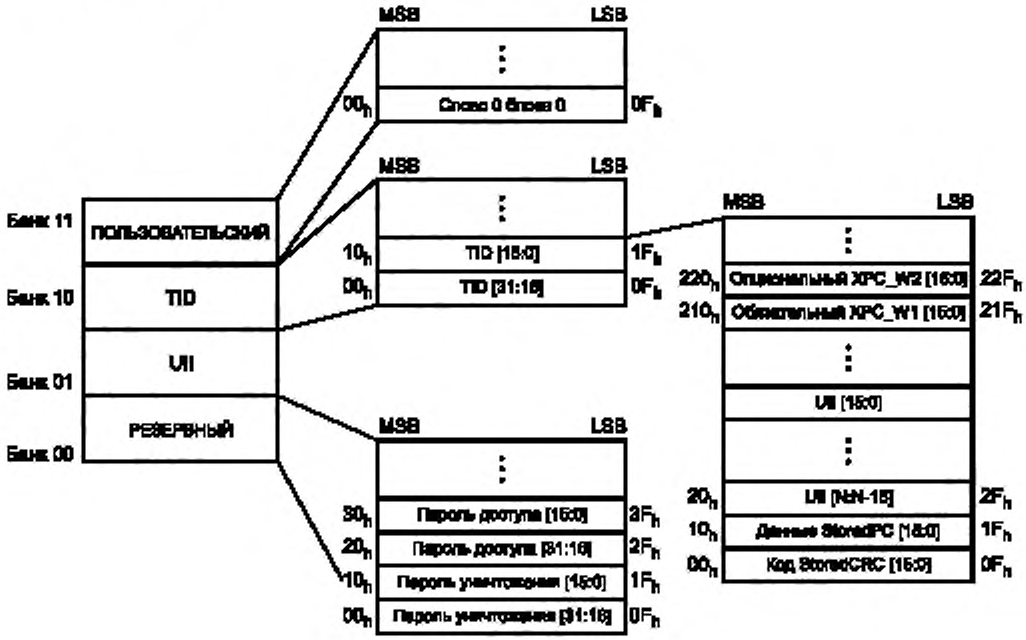



Рисунок 41 — Карта распределения памяти

Значения параметра MemBank определяются следующим образом:

$00_2$ Резервный

$01_2$ UII

$10_2$ TID

$11_2$ Пользовательский

При работе с одним банком логической памяти не должно быть доступа к памяти других банков.

Запись в память, подробно описанная в 6.3.3.4.9, предусматривает передачу 16-битовых слов от УСО к радиочастотной метке. Команда *Write* записывает 16 бит (т. е. одно слово) сразу, опционально используя защиту кодированием для сокрытия данных во время передачи R=>T. Опциональная команда *BlockWrite* записывает одновременно одно или более 16-битовых слов без использования защиты соединения кодированием. Опциональная команда *BlockErase* стирает одно или более 16-битовых слов одновременно. Команды *Write*, *BlockWrite* или *BlockErase* не должны изменять внутреннее состояние радиочастотной метки, находящейся в состоянии «Уничтожена» — постоянно не отвечающей, независимо от значения адреса памяти (достоверного или ошибочного) указанного в команде.

УСО может заблокировать, «постоянно заблокировать», разблокировать или «постоянно разблокировать» пароль уничтожения, пароль доступа, память UII, память TID и Пользовательскую память, и таким образом предотвратить или разрешить последующие изменения внутреннего состояния (по обстоятельствам). Радиочастотная метка опционально может иметь свою Пользовательскую память, разделенную на блоки; если это так, то УСО может «постоянно заблокировать» эти блоки. После блокирования или постоянного блокирования памяти ее можно вернуть в рабочее состояние. Описание блокировки и разблокировки памяти см. в 6.3.3.4.9, а описание ввода в действие радиочастотной метки см. в 6.3.3.4.10. Если пароли уничтожения и/или доступа заблокированы, то они применимы только по командам *Kill* и Access соответственно, при этом оба остаются незаписываемыми и нечитаемыми для любой другой команды. Блокировка или «постоянная блокировка» банков UII, TID или Пользовательской памяти делает заблокированную память незаписываемой, но оставляет доступной для считывания.

6.3.3.4.1.1 Резервная память

Резервная память должна содержать пароли уничтожения (см. 6.3.3.4.1.1.1) и/или доступа (см. 6.3.3.4.1.1.2), если в радиочастотной метке установлены пароли. Если радиочастотная метка не реализует паролей уничтожения или доступа, то она должна логически действовать, словно имеет пароли с нулевым значением, которые постоянно блокируют чтение/запись (см. 6.3.3.4.11.3.5), при этом физическая память, соответствующая Резервной памяти, не нужна.

6.3.3.4.1.1.1 Пароль уничтожения

Значение 32-битового пароля уничтожения размещается в банке резервной памяти, по адресу $00_h$ по $1F_h$, начиная со старшего значащего бита (MSB). По умолчанию его значение (непрограммируемое) — ноль. УСО может использовать пароль уничтожения для «ввода в действие» радиочастотной метки и/или «уничтожения» радиочастотной метки и перевода ее в не реагирующее состояние. Радиочастотная метка не должна выполнять операцию «ввода в действие» или «уничтожения», если ее пароль уничтожения имеет нулевое значение. Радиочастотная метка, которая не поддерживает операции, связанные с паролем уничтожения, представляется, как если бы она имела нулевой пароль уничтожения, не доступный для считывания/записи.

6.3.3.4.1.1.2 Пароль доступа

Значение 32-битового пароля доступа размещается в банке резервной памяти, по адресу $20_h$ по $3F_h$, начиная со старшего значащего бита (MSB). По умолчанию его значение (непрограммируемое) — ноль. Радиочастотная метка с ненулевым паролем доступа должна требовать УСО передачи этого пароля перед переходом в состояние «**Защищена**» (**secured**). Радиочастотная метка, которая не реализует операции, связанные с паролем доступа, представляется, как если бы она имела нулевой пароль доступа, не доступный для считывания/записи.

6.3.3.4.1.2 Память UII

Память UII содержит коды StoredCRC в памяти по адресам с $00_h$ по $0F_h$, информацию Stored PC по адресам с $10_h$ по $1F_h$, UII, начинающийся с $20_h$, первое слово XPC (XPC_W1) по адресам с $210_h$ по $21F_h$ и, опционально, второе слово XPC (XPC_W2) по адресам с $220_h$ по $22F_h$. Коды StoredCRC, данные StoredPC, UII слово или слова XPC должны храниться старшим значащим битом вперед (т. е. старший значащий бит UII размещается по адресу $20_h$).

Коды StoredCRC описаны в 6.3.3.4.1.2.1.

Данные StoredPC, как описано в 6.3.3.4.1.2.2, подразделяется на поле длины UII, расположенное в памяти по адресам с $10_h$ по $14_h$, индикатор Пользовательской памяти (UMI), расположенный по адресу $15_h$, индикатор XPC (XI), расположенный по адресу $16_h$, и идентификатор системы счисления (NSI), расположенный по адресам с $17_h$ по $1F_h$.

UII — это код, идентифицирующий объект, к которому прикреплена радиочастотная метка. UII для приложений EPCglobal™ описан в 6.3.3.4.1.2.3; *UII* для приложений ИСО описан в 6.3.3.4.1.2.4. УСО может подать команду *Select*, которая содержит в маске все или часть UII. УСО может подать команду *Ack*, которая заставит радиочастотную метку передать ее слово PC, XPC (если *XI* активирован), UII, и, при необходимости, код PacketCRC. В некоторых обстоятельствах радиочастотная метка может усекать свой ответ (см. 6.3.2.11.1.1). УСО может подать команду *Read* для считывания всего или части UII.

6.3.3.4.1.2.1 CRC-16 (код StoredCRC и код PacketCRC)

Все радиочастотные метки должны включать в себя реализацию кода StoredCRC. Радиочастотные метки также должны включать в себя реализацию кода PacketCRC.

При включении питания радиочастотная метка должна рассчитать CRC-16 по (а) данным StoredPC и (б) UII, заданным полем длины UII в данных StoredPC (см. 6.3.4.1.2.1), и должна сохранить рассчитанный код CRC-16 в памяти UII по адресам с $00_h$ по $0F_h$, старший значащий бит первый. Этот код CRC обозначается как код StoredCRC. Поскольку данные StoredPC и UII содержат целое число слов области памяти UI, радиочастотная метка рассчитывает коды StoredCRC по границам слов. Хотя при расчете кодов StoredCRC радиочастотные метки включают значение XI в данные StoredPC, при любом значении XI радиочастотная метка должна исключать слова XPC_W1 и XPC_W2 из расчета. Радиочастотные метки должны закончить расчет CRC-16 и распределение данных в памяти до окончания интервала $T_s$, показанного на рисунке 21. УСО может подать команду *Select*, которая содержит в маске все или часть кода StoredCRC. УСО может подать команду *Read*, что заставит радиочастотную метку передать ее код StoredCRC. Радиочастотные метки не должны пересчитывать свои коды StoredCRC в случае использования усеченного ответа (см. 6.3.3.4.11.1.1).

В ответ на команду *ACK* радиочастотная метка выполняет передачу слова Управления протоколом (*PC*) (либо данные StoredPC, либо данные PacketPC — см. 6.3.3.4.1.2.2), первое слово XPC (XPC_W1) и

опциональное второе слово XPC (XPC_W2), которые зависят от *XI* (см. 6.3.3.4.1.2.5), *UII* (см. 6.3.3.4.1.2.3 и 6.3.3.4.1.2.4), а также код CRC-16, только в случае, если затребован код PacketCRC, в соответствии с таблицей 42, приведенной ниже. В этом случае CRC-16 должен являться кодом PacketCRC, который радиочастотная метка должна динамически рассчитать по передаваемым: слову PC, слову или словам XPC (если поддерживаются) и UII. Независимо от того, пропускает радиочастотная метка CRC16 или определяет его при передаче, код PacketCRC должен быть представлен, как показано в таблице 42, в зависимости от величины XI радиочастотной метки и независимо от того, используется или не используется режим усечения ответа.

Таблица 42 — Данные радиочастотной метки и, если затребован, код PacketCRC, передаваемые в ответ на команду *ACK*

| XI | XEB | Усечение | Ответ радиочастотной метки | | | |
|---|---|---|---|---|---|---|
| | | | PC | XPC | UII | CRC-16 |
| 0 | 0 | Деактивирован | Данные StoredPC | Нет | Полный | В ответе нет CRC16 |
| 0 | 0 | Активирован | $00000_2$ | Нет | Усеченный | В ответе нет CRC16 |
| 0 | 1 | Деактивирован | Ошибочное значение | | | |
| 0 | 1 | Активирован | Ошибочное значение | | | |
| 1 | 0 | Деактивирован | Данные PacketPC | XPC_W1 | Полный | Код PacketCRC |
| 1 | 0 | Активирован | $00000_2$ | Нет | Усеченный | В ответе нет CRC16 |
| 1 | 1 | Деактивирован | Данные PacketPC | Оба XPC_W1 и XPC_W2 | Полный | Код PacketCRC |
| 1 | 1 | Активирован | $00000_2$ | Нет | Усеченный | В ответе нет CRC16 |

Примечания

1 XI — это поразрядное логическое ИЛИ (OR) 16-битовых XPC_W1, а XEB — старший значащий бит (бит $F_h$) XPC_W1, так что если XEB = 1, то XI = 1.

2 Код StoredCRC может отличаться от кода PacketCRC, если память радиочастотной метки была запрограммирована и радиочастотная метка не подвергалась сбросу при выключении-включении питания.

Радиочастотная метка должна передать свой CRC старшим значащим битом вперед в начале передачи, независимо от типа CRC.

Если XI активирован, то код PacketCRC радиочастотной метки отличается от ее кода StoredCRC.

Как требует 6.3.3.3.1.5, УСО должно проверить, используя переданный меткой CRC-16, целостность полученного слова *PC*, первого слова XPC (XPC_W1), опционально, второго слова XPC (XPC_W2) и UII.

6.3.3.4.1.2.2 Слово Управления протоколом (PC)

Все радиочастотные метки должны включать в себя реализацию элемента данные StoredPC, чьи поля, включая длину UII, UMI, XI и NSI, должны соответствовать указанным ниже. Радиочастотные метки должны также включать в себя реализацию элемента данные PacketPC, отличающегося от элемента данные StoredPC полем длина UII. Информация о типе PC (данные StoredPC или данные PacketPC), которую радиочастотная метка передает в ответе на команду ACK, должна быть определена согласно таблице 42.

Данные StoredPC должны размещаться в памяти UII по адресам с $10_h$ по $1F_h$ со значениями битов, определенными следующим образом:

- Биты $10_h$ — $14_h$: длина UII, который радиочастотная метка передает, в словах:
  - $00000_2$: Нулевые слова.
  - $00001_2$: Одно слово (адреса с $20_h$ по $2F_h$ в памяти UII).
  - $00010_2$: Два слова (адреса с $20_h$ по $3F_h$ в памяти UII).
  - .
  - .
  - .
  - $11101_2$: 29 слов (адреса с $20_h$ по $1EF_h$ в памяти UII).

- Максимальное значение поля длина UII в данных StoredPC должно быть $11101_2$ (разрешает 464-битовое UII). Радиочастотная метка должна игнорировать команду Write или BlockWrite для данных StoredPC, если поле длина UII превышает $11101_2$, и вместо этого следует выполнить передачу кода ошибки (см. приложение Е).

- Бит $15_h$: Индикатор Пользовательской памяти (UMI). Если бит $15_h$ деактивирован, то радиочастотная метка либо не использует Пользовательскую память, либо память пользователя не содержит информацию. Если бит $15_h$ активирован, то память пользователя содержит информацию. Радиочастотная метка может реализовать UMI, используя Метод 1 или Метод 2, описанные ниже, если только радиочастотная метка не поддерживает функции «постоянной блокировки» и/или «ввода в действие». В этом случае радиочастотная метка должна использовать Метод 1. В случае когда поддерживается обязательный «ввод в действие» с активированным третьим младшим битом Recom и не поддерживается «постоянная блокировка», вместо Метода 1 можно использовать Метод 2.

  - Метод 1: Радиочастотная метка определяет UMI. При включении питания и перед расчетом кода StoredCRC она должна произвести логическое ИЛИ битов с $03_h$ по $07_h$ Пользовательской памяти и занести полученное значение в бит $15_h$. Радиочастотная метка должна учесть полученное значение UMI в коде StoredCRC, рассчитанное при включении питания (см. 6.3.3.4.1.2.1). Если УСО модифицирует любой из битов с $03_h$ по $07_h$ Пользовательской памяти, то радиочастотная метка должна заново определить и перезаписать новое значение UMI в бит $15_h$. Если при «вводе в действие» метки память пользователя оказывается недоступна (см. 6.3.3.4.10), то радиочастотная метка должна переопределить и перезаписать UMI в бит $15_h$. После перезаписи UMI код StoredCRC может оказаться неверными до тех пор, пока УСО выполняет текущий энергетический цикл. УСО не должно иметь возможность записи значения UMI — если УСО записывает данные StoredPC, радиочастотная метка игнорирует значение $15_h$ бита данных, которое передает УСО.
  - Метод 2: УСО записывает UMI. Если УСО записывает нулевое значение в биты с $03_h$ по $07_h$ Пользовательской памяти, то бит $15_h$ должен быть сброшен. Если УСО записывает ненулевое значение в биты с $03_h$ по $07_h$ Пользовательской памяти, то бит $15_h$ должен быть активирован. Если УСО блокирует или «постоянно блокирует» память UII, то оно должно также блокировать или «постоянно блокировать», соответственно, слово, расположенное по адресу $00_h$ Пользовательской памяти, и наоборот. Это последнее требование гарантирует, что ситуация при которой память пользователя ранее содержала данные, но они были впоследствии стерты, не станет для радиочастотной метки причиной неверной индикации наличия Пользовательской памяти, и наоборот.

- Бит $16_h$: Индикатор XPC_W1 (XI). Если бит $16_h$ сброшен, то XPC_W1 имеет нулевое значение, и в этом случае радиочастотная метка во время инвентаризации должна выполнить передачу данных StoredPC, но не XPC_W1 (см. 6.3.3.4.1.2). Если бит $16_h$ установлен, то один или более битов из XPC_W1 имеют ненулевые значения, показывая, что радиочастотная метка была предварительно «введена в действие» (см. 6.3.3.4.1.2.5).

В последнем случае радиочастотная метка должна выполнить передачу своего XPC_W1 сразу же за данными PacketPC и перед UII в течение инвентаризации.

При включении питания и перед расчетом кодов StoredCRC радиочастотная метка должна произвести побитовое логическое ИЛИ XPC_W1 и занести полученное значение в бит $16_h$ (т. е. в XI). Радиочастотная метка должна учитывать полученное значение XI при определении кода StoredCRC. Если УСО «вводит в действие» радиочастотную метку (см. 6.3.3.4.10), то радиочастотная метка должна пересчитать и перезаписать свое XI в бит $16_h$. После перезаписи XI код StoredCRC может оказаться неверным до тех пор, пока УСО выполняет текущий энергетический цикл. УСО не должно иметь возможность записи значения XI — если УСО записывает данные StoredPC, радиочастотная метка игнорирует значение $16_h$ бита данных, которое передает УСО.

- Биты $17_h$ — $1F_h$: Идентификатор системы счисления (NSI). Старший значащий бит NSI хранится в памяти по адресу $17_h$. Если бит $17_h$ содержит логический 0, то пользовательское приложение руководствуется стандартом EPCglobal™, а биты от $18_h$ до $1F_h$ должны быть определены в *[2]*. Если бит $17_h$ содержит логическую 1, то пользовательское приложение определяется как «Приложение по ИСО», а биты от $18_h$ до $1F_h$ должны содержать полное значение AFI, определенное в *[6] и [5]*. По умолчанию значения для битов от $18_h$ до $1F_h$ $00000000_2$.

По умолчанию значение данных StoredPC (незапрограммированное) должно быть $0000_h$.

Если УСО изменит длину UII (через операцию записи в память) и если радиочастотной метке для дальнейшей передачи данных понадобится новое значение длины UII, то она должна записать новое поле длины UII в первые 5 бит данных StoredPC радиочастотной метки.

Примечание — После изменения поля длины UII УСО должно выполнить энергетический цикл сброса радиочастотной метки, чтобы гарантировать правильность значения кода StoredCRC.

Если УСО попыталось записать в поле длины UII значение, которое не поддерживается радиочастотной меткой, метка должна передать код ошибки (см. приложение Е) в первых 5 битах слова PC радиочастотной метки.

Данные PacketPC отличаются от данных StoredPC полем длины UII, радиочастотная метка должна настраивать это поле в соответствии с длиной переданных данных, следующих за словом PC. В частности, если XI деактивирован, а XEB не активирован, то радиочастотная метка выполняет передачу слова XPC_W1 перед UII, так что радиочастотная метка должна добавить единицу (т. е. инкрементировать) поле длина UII. Если XI и XEB активированы, то радиочастотная метка выполняет передачу обоих слов XPC_W1 и XPC_W2 перед UII, так что радиочастотная метка должна добавить два (т. е. инкрементировать дважды) в поле длины UII. Поскольку радиочастотные метки, поддерживающие функционирование XPC, имеют максимальное значение поля длины UII $11101_2$, двойной инкремент увеличит значение до $11111_2$. Радиочастотная метка не должна ни при каких обстоятельствах допускать переполнения поля длины UII до $00000_2$. Следует помнить, что инкремент или двойной инкремент поля длины UII не изменяет значение, хранимое в битах от $10_h$ до $14_h$ памяти UII; точнее, радиочастотная метка инкрементирует поле длина UII при передаче данных PacketPC, но сохраняет содержимое памяти неизменным.

Если УСО не поддерживает получение XPC_W2, радиочастотная метка дает ответ с активированным XEB, а затем УСО должно обработать ответ радиочастотной метки, как будто его проверка целостности CRC-16 была неуспешной.

Во время обработки ответов радиочастотная метка подставляет $\mathbf{00000_2}$ в слово PC.

6.3.3.4.1.2.3 UII Приложения EPCglobal™

Структура UII для приложения EPCglobal™ должна быть такой, как определено в документе *[2]*.

6.3.3.4.1.2.4 UII для приложения ИСО

Структура UII для приложения ИСО должна быть такой, как определено в *[3]*.

6.3.3.4.1.2.5 Слово расширенного управления протоколом (XPC). Одно (обязательно) или два (опционально)

В радиочастотной метке должно быть реализовано слово XPC_W1, логически расположенное по адресам с $210_h$ по $21F_h$ памяти UII. В радиочастотной метке может быть дополнительно реализовано слово XPC_W2, логически расположенное по адресам с $220_h$ по $22F_h$ памяти UII. Эти слова XPC должны быть длиной точно 16 бит и храниться старшим значащим битом вперед. Если радиочастотная метка не поддерживает XPC_W2, то указанное для него место расположения в памяти не используется.

Радиочастотная метка не должна располагать любые элементы данных, не являющиеся XPC, в памяти UII по адресам с $210_h$ по $22F_h$ включительно. Это требование должно применяться ко всем радиочастотная меткам — которые поддерживают слово XPC_W2 и которые его не поддерживают.

Если радиочастотная метка поддерживает XPC_W2, то при включении питания она должна выполнить поразрядное логическое ИЛИ XPC_W2 и занести полученное значение в бит $210_h$ памяти UII (т. е., в старший значащий бит XPC_W1). Бит $210_h$ указывает на бит расширения (XEB) XPC. Если радиочастотная метка не поддерживает XPC_W2, то значение XEB должно быть нулевым.

Далее до конца этого пункта 6.3.3.4.1.2.5 предполагается, что радиочастотная метка поддерживает слова XPC_W1 (обязательное) и XPC_W2 (справочное).

Когда этот документ ссылается на 3 младших значащих бита XPC_W1, это означает адреса $21D_h$, $21E_h$ и $21F_h$ памяти UII. 3 младших значащих бита XPC_W1 указывают, была ли и каким образом была «введена в действие» радиочастотная метка.

Для не «введенных в действие» радиочастотных меток 3 младших значащих бита XPC_W1 должны быть нулевыми. Радиочастотная метка записывает ненулевое значение в один или более из этих 3 младших значащих битов во время «ввода в действие» (см. 6.3.3.4.10). Все другие биты XPC_W1, а именно адреса памяти UII с $210_h$ по $21C_h$ включительно, как и все биты в XPC_W2, должны быть RFU*

---

* *Зарезервировано для будущего использования. См. 5.2.*

и иметь нулевые значения. Вендоры и конечные пользователи радиочастотной метки не должны использовать эти биты RFU для собственных целей.

Три младших значащих бита XPC_W1 не могут быть записаны с помощью команд *Write* или *BlockWrite* и не могут быть очищены с помощью команды *BlockErase*. Они могут быть установлены только командой *Kill* и означают, что радиочастотная метка установила эти биты потому, что приняла достоверную команду *Kill* с установленными битами «ввода в действие» (см. 6.3.3.4.11.3.4). Радиочастотная метка не должна ничего записывать в 3 младших значащих бита XPC_W1, за исключением случая «ввода в действие».

Если радиочастотная метка принимает команду *Write, BlockWrite* и *BlockErase*, которая пытается записать XPC_W1, то она отвечает кодом ошибки (см. приложение Е).

УСО может дать команду *Select* (см. 6.3.3.4.11.1) со значением параметра Mask, которое покрывает все или часть слов XPC_W1 и/или XPC_W2. Например, Mask может иметь значение $\mathbf{000}_2$ для 3 младших значащих битов XPC_W1, в этом случае радиочастотная метка не будет соответствовать «введенной в действие».

УСО может считывать значения XPC_W1 и XPC_W2 радиочастотной метки с помощью команды *Read* (см. 6.3.3.4.11.3.2).

Бит $E_h$ XPC_W1 (расположен в памяти UII по адресу $211_h$) зарезервирован для использования в качестве индикатора функциональности протокола.

Следующее кодирование определяет общее соответствие между 3 младшими значащими битами *XPC_W1* и статусом «введена в действие» радиочастотной метки. В таблице 43 приведено подробное описание соответствия между этими 3 разрядами и статусом метки «введена в действие»:

- **Установленный младший значащий бит** (расположен в памяти UII по адресу $21F_h$) показывает, что «постоянная блокировка» Пользовательской памяти запрещена, и любые блоки Пользовательской памяти, которые были ранее «постоянно блокированы», больше не блокированы. Активированный младший значащий бит также показывает, что команда *BlockPermalock* запрещена. Если в радиочастотной метке перед «вводом в действие» не реализована «постоянная блокировка» блоков Пользовательской памяти, тогда «постоянная блокировка» будет оставаться отключенной.

- **Установленный второй младший значащий бит** (располагается в памяти UII по адресу 21Eh) показывает, что память пользователя была сделана недоступной. Второй младший значащий бит имеет больший приоритет по сравнению с самым младшим значащим битом — в случае если оба разряда активированы, память пользователя будет недоступна.

- **Установленный третий младший значащий бит** (располагается в памяти UII по адресу 21Dh) показывает, что радиочастотная метка имеет разблокированные банки памяти UII, TID и Пользовательской памяти. Кроме того, в этом случае радиочастотная метка имеет незаблокированные для записи области паролей для уничтожения (kill) и доступа (access), и сохраняет статус Read/Lock (считывания/блокировки) паролей уничтожения и доступа в состоянии до «ввода в строй». Если пароли уничтожения и доступа были не читаемы до «ввода в действие», и УСО попытается их считывать, радиочастотная метка передаст код ошибки (см. приложение Е). Следует отметить, что если сегменты или банки Пользовательской памяти радиочастотной метки были записаны и заблокированы при их производстве, они не могут быть разблокированы независимо от «ввода в действие». Также следует отметить, что УСО может впоследствии опять заблокировать любые банки Пользовательской памяти, которые были разблокированы при «вводе в действие».

Таблица 43 — Младшие значащие биты XPC и статус «введена в действие» радиочастотной метки

| МЗР слова XPC | Статус радиочастотной метки | Примечание |
|---|---|---|
| $000_2$ | 1. Радиочастотная метка «введена в действие» | 1. Бит XI радиочастотной метки не установлен |
| $001_2$ | 1. «Постоянная блокировка» блока отменена, и любые блоки Пользовательской памяти, которые предварительно имели «постоянную блокировку», больше ее не имеют.<br>2. Команда *BlockPermalock* отменена | 1. Биты блокировки — единственный определяющий фактор состояния блокировки блоков Пользовательской памяти (6.3.3.4.11.3.5) |

*Продолжение таблицы 43*

| МЗР слова XPC | Статус радиочастотной метки | Примечание |
|---|---|---|
| $010_2$ | 1. Банк Пользовательской памяти сделан недоступным.<br>2. Радиочастотная метка, чьи младшие значащие биты XPC_W1 имеют состояние $010_2$, работает так же, как если младшие биты XPC_W1 имеют состояние $011_2$ | 1. Радиочастотная метка сбрасывает свой бит UMI.<br>2. Пользовательская память недоступна, так что состояние «постоянная блокировка» блока и команда *BlockPermalock* отменены |
| $011_2$ | 1. Банк Пользовательской памяти сделан недоступным.<br>2. Радиочастотная метка, чьи младшие значащие биты XPC_W1 имеют состояние $011_2$, работает так же, как если младшие биты XPC_W1 имеют состояние $010_2$ | 1. Радиочастотная метка деактивирует свой бит UMI.<br>2. Пользовательская память недоступна, так что состояние «постоянная блокировка» блока и команда *BlockPermalock* отменены |
| $100_2$ | 1. Банки UII, TID и Пользовательской памяти разблокированы.<br>2. Пароли уничтожения и доступа разблокированы для записи.<br>3. Статус Read/Lock паролей уничтожения и доступа должны иметь такое же состояние, как перед «вводом в действие» | 1. Если радиочастотная метка поддерживает «постоянную блокировку» блока, то команда *BlockPermalock* остается действительной.<br>2. Любые блоки Пользовательской памяти, которые предварительно имели «постоянную блокировку», остаются в этом состоянии, и наоборот.<br>3. Если сегменты или банки Пользовательской памяти радиочастотной метки были записаны и заблокированы при их производстве, они не могут быть разблокированы независимо от «ввода в действие».<br>4. Если пароли уничтожения и доступа были не читаемы до «ввода в действие», и УСО попытается их считывать, радиочастотная метка передаст код ошибки.<br>5. Пароли уничтожения и/или доступа, так же как один или несколько блоков и/или банков памяти, можно изменить и/или повторно заблокировать после «ввода в действие» |
| $101_2$ | 1. «Постоянная блокировка» блока отменена, и любые блоки Пользовательской памяти, которые предварительно имели постоянную блокировку, больше ее не имеют.<br>2. Команда *BlockPermalock* отменена.<br>3. Банки UII, TID и Пользовательской памяти разблокированы.<br>4. Пароли уничтожения и доступа разблокированы для записи.<br>5. Статус Read/Lock паролей уничтожения и доступа должны иметь такое же состояние, как перед «вводом в действие» | 1. Биты блокировки — единственный определяющий фактор состояния блокировки блоков Пользовательской памяти (6.3.3.4.11.3.5).<br>2. Если сегменты или банки Пользовательской памяти радиочастотной метки были записаны и заблокированы при их производстве, они не могут быть разблокированы независимо от «ввода в действие».<br>3. Если пароли уничтожения и доступа были не читаемы до «ввода в действие», и УСО попытается их считывать, радиочастотная метка передаст код ошибки.<br>4. Пароли уничтожения и/или доступа, также как один или несколько блоков и/или банков памяти можно изменить и/или повторно заблокировать после «ввода в действие» |
| $110_2$ | 1. Области памяти UII, TID разблокированы.<br>2. Пароли для уничтожения и доступа разблокированы для записи.<br>3. Статус Read/Lock паролей уничтожения и доступа должны иметь такое же состояние, как перед «вводом в действие».<br>4. Банк Пользовательской памяти сделан недоступным. | 1. Если сегменты или банки Пользовательской памяти радиочастотной метки были записаны и заблокированы при их производстве, они не могут быть разблокированы независимо от «ввода в действие».<br>2. Если пароли уничтожения и доступа были не читаемы до «ввода в действие» и УСО попытается их считывать, радиочастотная метка передаст код ошибки. |

*Окончание таблицы 43*

| МЗР слова XPC | Статус радиочастотной метки | Примечание |
|---|---|---|
| | 5. Радиочастотная метка, чьи младшие значащие биты XPC_W1 имеют состояние $110_2$, работает так же, как если младшие биты XPC_W1 имеют состояние $111_2$ | 3. Пароли уничтожения и/или доступа, так же как один или несколько блоков и/или банков памяти, можно изменить и/или повторно заблокировать после «ввода в действие».<br>4. Радиочастотная метка сбрасывает свой бит UMI<br>5. Память пользователя недоступна, так что «постоянная блокировка» блоков и команда *BlockPermalock* отменены |
| $111_2$ | 1. Области памяти UII, TID разблокированы.<br>2. Пароли для уничтожения и доступа разблокированы для записи.<br>3. Статус Read/Lock паролей уничтожения и доступа должны иметь такое же состояние, как перед «вводом в действие».<br>4. Банк Пользовательской памяти сделан недоступным.<br>5. Радиочастотная метка, чьи младшие значащие биты XPC_W1 имеют состояние $111_2$, работает так же, как если младшие биты XPC_W1 имеют состояние $110_2$ | 1. Если сегменты или банки Пользовательской памяти радиочастотной метки были записаны и заблокированы при их производстве, они не могут быть разблокированы независимо от «ввода в действие».<br>2. Если пароли уничтожения и доступа были не читаемы до «ввода в действие» и УСО попытается их считывать, радиочастотная метка передаст код ошибки.<br>3. Пароли уничтожения и/или доступа, также как один или несколько блоков и/или банков памяти можно изменить и/или повторно заблокировать после «ввода в действие».<br>4. Радиочастотная метка сбрасывает свой бит UMI.<br>5. Память пользователя недоступна, так что «постоянная блокировка» блоков и команда *BlockPermalock* отменены |

6.3.3.4.1.3 Память TID

Память TID, располагающаяся по адресам с $00_h$ по $07_h$, должна содержать одно из двух значений идентификатора класса по *ГОСТ Р ИСО/МЭК 15963*: $E0_h$ или $E2_h$. Память TID, располагающаяся выше адреса $07_h$, должна быть определена согласованно с предписаниями органов регистрации по этому значению идентификатора класса и должна содержать, как минимум, достаточную идентифицирующую информацию для УСО для однозначного опознавания команд пользователя и/или дополнительных возможностей, поддерживаемых радиочастотной меткой. Память TID также может содержать специфические данные радиочастотной метки и продавца (например, серийный номер радиочастотной метки).

Примечание — Изготовитель радиочастотной метки устанавливает значение идентификатора класса (т. е., $E0_h$ или $E2_h$), которое определено ему органом регистрации по *ГОСТ Р ИСО/МЭК 15963*. Идентификатор класса не определяет приложение. Если идентификатор класса имеет значение $E0_h$, память TID, расположенная по адресам с $08_h$ по $0F_h$, содержит 8-битовый идентификатор изготовителя, память TID, расположенная по адресам с $10_h$ по $3F_h$ содержит 48-битовый серийный номер радиочастотной метки (присваиваемый изготовителем радиочастотной метки), объединение дает 64-битовый идентификатор радиочастотной метки (т. е., память TID с $00_h$ по $3F_h$) — уникальный среди всех классов радиочастотных меток и определенный в *ГОСТ Р ИСО/МЭК 15693*, при этом память TID «постоянно блокирована» во время производства. Если идентификатор класса имеет значение $E2_h$, память TID, расположенная по адресу $08_h$ содержит XTID; Память TID, расположенная по адресам с $09_h$ по $13_h$, содержит 11-битовый идентификатор маски разработчика радиочастотной метки (получаемый в органе регистрации), память TID, расположенная по адресам с $14_h$ по $1F_h$, содержит определяемый вендором 12-битовый номер модели радиочастотной метки, а использование памяти TID выше $1F_h$ определено в *[2]*.

6.3.3.4.1.4 Пользовательская память

Радиочастотная метка может содержать Пользовательскую память. Пользовательская память позволяет хранить данные пользователя.

Если Пользовательская память радиочастотной метки кроме того непрограммируемая, то 5 младших значащих битов первого байта Пользовательской памяти (т. е., память, адресуемая с $03_h$ по $07_h$) должны иметь значение по умолчанию $00000_2$.

При «вводе в действие» УСО может дать команду радиочастотной метке сделать ее Пользовательскую память недоступной, что приводит к тому, что весь банк памяти становится нечитаемым, неза-

писываемым и невыбираемым. Радиочастотная метка с недоступной Пользовательской памятью должна функционировать так, как будто ее банк Пользовательской памяти больше не существует.

6.3.3.4.1.4.1 Пользовательская память для приложения EPCglobal™

Если радиочастотная метка имеет пользовательскую память, то ее содержимое формируется в соответствии со стандартами «EPCglobal™ Стандарты данных радиочастотной метки EPC» (версия 1.3 и выше).

6.3.3.4.1.4.2 Память пользователя для приложения ISO

Если радиочастотная метка имеет память пользователя, то ее содержимое формируется в соответствии с *[8]* и *[3]*.

6.3.3.4.2 Методы ASK и PJM: Сеансы и флаги **inventoried**

Примечание — УСО выбирает один из двух сеансов и проводит инвентаризацию радиочастотных меток в выбранном сеансе. УСО и связанное множество радиочастотных меток действуют в одном и только одном сеансе в течение цикла инвентаризации (как было определено выше). Для каждого сеанса радиочастотные метки поддерживают соответствующий флаг инвентаризации. Сеансы позволяют радиочастотным меткам хранить их текущее состояние инвентаризации отдельно для каждого из двух возможных чередующихся процессов инвентаризации, используя независимый флаг **inventoried** для каждого процесса.

УСО должно реализовывать поддержку двух обязательных сеансов (обозначаемых S0 и S2) и двух дополнительных сеансов (обозначаемых S1 и S3). Радиочастотные метки должны участвовать в одном и только одном сеансе в течение цикла инвентаризации. Два или более УСО могут использовать сеансы для независимой инвентаризации общего множества радиочастотных меток. Концепция сеанса иллюстрируется на рисунке 42.

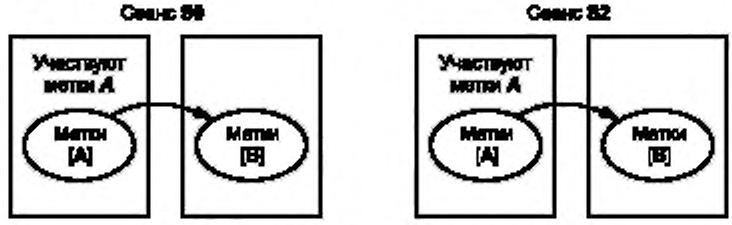



Рисунок 42 — Диаграмма сеансов

Радиочастотные метки должны сохранять независимый флаг **inventoried** для каждого сеанса. Каждый из двух обязательных флагов **inventoried** имеет два значения, обозначаемые A и B. Радиочастотные метки, участвующие в цикле инвентаризации в одном сеансе, не должны использовать или модифицировать флаг инвентаризации для других сеансов. Флаги **inventoried** являются для радиочастотной метки только средством, обеспечивающим отдельное и независимое выполнение данного сеанса инвентаризации; все другие ресурсы радиочастотной метки используются совместно всеми сессиями.

Сеансы обеспечивают возможность для радиочастотных меток учитывать раздельно и независимо флаги **inventoried** для каждой из нескольких инвентаризаций.

После захвата радиочастотной метки УСО может подать команду, по которой радиочастотная метка установит свой флаг **inventoried** для этого сеанса (A→B).

В цикле инвентаризации отвечают только радиочастотные метки с флагом **inventoried** A.

Примечание — Переключение с B → A может быть достигнуто выбором специальной команды. (Reset).

Следующий пример иллюстрирует, как два УСО могут использовать сеансы и флаги **inventoried** для независимой и полной инвентаризации общего множества радиочастотных меток на основе разделения времени на интервалы:

- УСО № 1 включается, затем:
- инициирует цикл инвентаризации, во время которого оно захватывает и переводит метки в сеансе S0 из значения A в B,
- выключает питание.
- УСО № 2 включается, затем:
- инициирует цикл инвентаризации, во время которого оно и переводит метки в сеансе S2 из значения A в B,
- выключает питание.

Этот процесс повторяется до тех пор, пока УСО № 1 не разместит все радиочастотные метки в сеансе S0 в B. Аналогично, пока УСО № 2 не разместит все радиочастотные метки в сеансе S2 в B.

Примечание — Радиочастотные метки сохраняют раздельно значения флага **inventoried** для каждого из двух обязательных сеансов.

Флаги **inventoried** радиочастотной метки должны иметь времена сохранения значения, указанные в таблице 44. Радиочастотная метка должна включаться со своим флагом **inventoried**, заданным следующим образом:

- Флаг **inventoried** S0 должен быть установлен в A.
- Флаг **inventoried** S1 (необязательный) должен быть установлен либо в A, либо в B, в зависимости от его сохраненного значения, если только с момента установки флага не прошло больше времени, чем его время сохранения, в этом случае радиочастотная метка должна включаться с флагом инвентаризации S1, установленным в A. Поскольку флаг **inventoried** S1 (необязательный) не обновляется автоматически, он может перейти из значения B в A, даже когда радиочастотная метка включена.
- Флаг **inventoried** S2 должен быть установлен либо в A, либо в B в зависимости от его сохраненного значения, если только с момента снятия питания с радиочастотной метки не прошло времени больше, чем время сохранения, в этом случае радиочастотная метка должна включаться с флагом **inventoried** S2, установленным в A.
- Флаг **inventoried** S3 (необязательный) должен быть установлен либо в A, либо в B в зависимости от его сохраненного значения, если только с момента снятия питания с радиочастотной метки не прошло времени больше, чем время сохранения, в этом случае радиочастотная метка должна включаться с флагом **inventoried** S3, установленным в A.

Радиочастотная метка должна быть способна устанавливать любой из ее флагов **inventoried** либо в значение A, либо в B за время 2 мс или менее, независимо от исходного значения флага. Радиочастотная метка должна обновлять свои флаги S2 и S3, будучи включенной, подразумевая, что всякий раз, когда радиочастотная метка лишается питания, ее флаги **inventoried** S2 и S3 должны иметь время сохранения, указанное в таблице 44.

Таблица 44 — Флаги радиочастотной метки и время их сохранения

| Флаг | Требуемое время сохранности |
|---|---|
| Флаг **inventoried** S0 (обязательный) | На радиочастотную метку подано питание: Не определено.<br>Радиочастотная метка не запитана: Нет |
| Флаг **inventoried** S1 (дополнительный) | На радиочастотную метку подано питание:<br>Диапазон номинальной температуры: 500 мс < продолжительность < 5 с<br>Расширенный температурный диапазон: Не определено<br>Радиочастотная метка а не запитана<br>Диапазон номинальной температуры: 500 мс < продолжительность < 5 с<br>Расширенный температурный диапазон: Не определено |
| Флаг **inventoried** S2 (дополнительный) | На радиочастотную метку подано питание: Не определено<br>Радиочастотная метка а не запитана<br>Диапазон номинальной температуры: 2 с < продолжительность<br>Расширенный температурный диапазон: Не определено |
| Флаг **inventoried** S3 (дополнительный) | На радиочастотную метку подано питание: Не определено<br>Радиочастотная метка а не запитана<br>Диапазон номинальной температуры: 2 с < продолжительность<br>Расширенный температурный диапазон: Не определено |
| Флаг **Selected** (**SL**) | На радиочастотную метку подано питание: Не определено<br>Радиочастотная метка а не запитана<br>Диапазон номинальной температуры: 2 с < продолжительность<br>Расширенный температурный диапазон: Не определено |

Примечания

1 Диапазон номинальной температуры от минус 25 °C до 40 °C (см. расширенный диапазон температуры).

2 Расширенный температурный диапазон от минус 40 °C до 65 °C (см. диапазон номинальной температуры).

6.3.3.4.3 Методы ASK и PJM: Флаг **Selected** (**SL**)

В радиочастотной метке должен быть реализован флаг **Selected SL**, который УСО может установить или снять с помощью команды *Select*. Параметр Select в команде *BeginRound* обеспечивает УСО доступ к инвентаризируемым меткам, имеющим либо установленный, либо сброшенный флаг **SL** (т. е. далее **SL** или **~SL**), либо игнорирует этот флаг и инвентаризируемые метки независимо от значения **SL**. **SL** не принадлежит к какому-либо определенному сеансу; **SL** применяется ко всем радиочастотным меткам независимо от сеанса.

Флаг **SL** радиочастотной метки должен иметь время сохранения, указанное в таблице 44. Радиочастотная метка должна включаться либо с установленным, либо со сброшенным флагом **SL**, в зависимости от сохраненного значения, если только с момента снятия питания с радиочастотной метки не прошло времени больше, чем время сохранения **SL**, в этом случае радиочастотная метка должна активироваться со сброшенным флагом **SL** (**~SL**). Радиочастотная метка может установить или сбросить флаг **SL** за 2 мс или менее, независимо от исходного значения. Радиочастотная метка должна обновлять свой флаг **SL**, будучи включенной, подразумевая, что всякий раз, когда радиочастотная метка теряет питание, ее флаг **SL** должен иметь время сохранения, указанное в таблице 44.

6.3.3.4.4 Методы ASK и PJM: Состояния метки и счетчик слота

В радиочастотных метках должны быть реализованы состояния и счетчик слота, как показано на рисунке 43. Приложение C показывает таблицы переходов связанных состояний; приложение D показывает таблицы соответствующих команд ответов.

Примечание — Протокол арбитража системы управления коллизиями радиочастотной метки использует общеизвестный и проверенный на практике алгоритм слотирования Aloha. Алгоритм слотирования Aloha является усовершенствованным по сравнению с простым алгоритмом Aloha. Он требует, чтобы интервал был разбит на слоты, разделенные маркерами слота. Вероятность того, что радиочастотная метка выберет временной интервал, равна 1/*n* (где *n* — число временных интервалов, равное максимальному значению, описываемому *Q*-битовым случайным числом). Этот алгоритм гарантирует, что каждая радиочастотная метка, если выбрана, ответит один и только один раз в течение цикла инвентаризации.

***Пример — Для Q = 3 максимальное число слотов 8, Q — 3-битовое случайное число в диапазоне от 0 до 7, для каждого слота вероятность ответа конкретной радиочастотной метки 1/8.***

6.3.3.4.4.1 Состояние **ready** (**готова**)

Радиочастотная метка должна включать в себя реализацию состояния **ready**. Состояние **ready** можно рассматривать как "состояние ожидания сигнала" для включенных радиочастотных меток, которые не «уничтожены» и не принимают в настоящее время участия в цикле инвентаризации. После возбуждения радиочастотного поля радиочастотная метка, которая не «уничтожена», должна перейти в состояние **ready**. Радиочастотная метка должна находиться в состоянии **ready** до тех пор, пока не примет команду *BeginRound* (см. 6.3.3.4.11.2.1), чей параметр inventoried (для сеанса, указанного в *BeginRound*) и параметр *Select* соответствует текущим значением ее флага. Соответствующие признакам радиочастотные метки должны сформировать *Q*-битовое число из своего RNG* (см. 6.3.3.4.5), загрузить это число в свой счетчик слота, и перейти в состояние **arbitrate**, если число не нулевое, или в состояние **reply**, если число нулевое. Если радиочастотная метка в любом состоянии, кроме отключенного **killed**, лишается питания, то она должна вернуться в состояние **ready** после получения энергии.

6.3.3.4.4.2 Состояние **arbitrate** (**арбитраж**)

Радиочастотная метка должна включать в себя реализацию состояния **arbitrate**. Состояние **arbitrate** можно рассматривать как «состояние ожидания сигнала» для радиочастотных меток, которые принимают участие в текущем цикле инвентаризации, и чьи счетчики слота (см. 6.3.3.4.4.8) имеют ненулевое значение. Радиочастотная метка в состоянии **arbitrate** должна декрементировать счетчик слота каждый раз, как принимает команду *NextSlot* (см. 6.3.3.4.11.2.3), у которой параметр сеанса session соответствует выполняемому в настоящее время сеансу инвентаризации. Когда ее счетчик слота достигнет значения $0000_h$, она должна перейти в состояние **reply**. Радиочастотные метки, которые вернулись в состояние **arbitrate** (например, из состояния **reply**) при значении слота $0000_h$, должны декрементировать свой счетчик слота от $0000_h$ до $7FFF_h$ по следующей команде NextSlot (с соответствующим параметром сеанса) и, поскольку их значение слота теперь ненулевое, должны оставаться в состоянии **arbitrate**.

---

* *Генератор псевдослучайных чисел.*

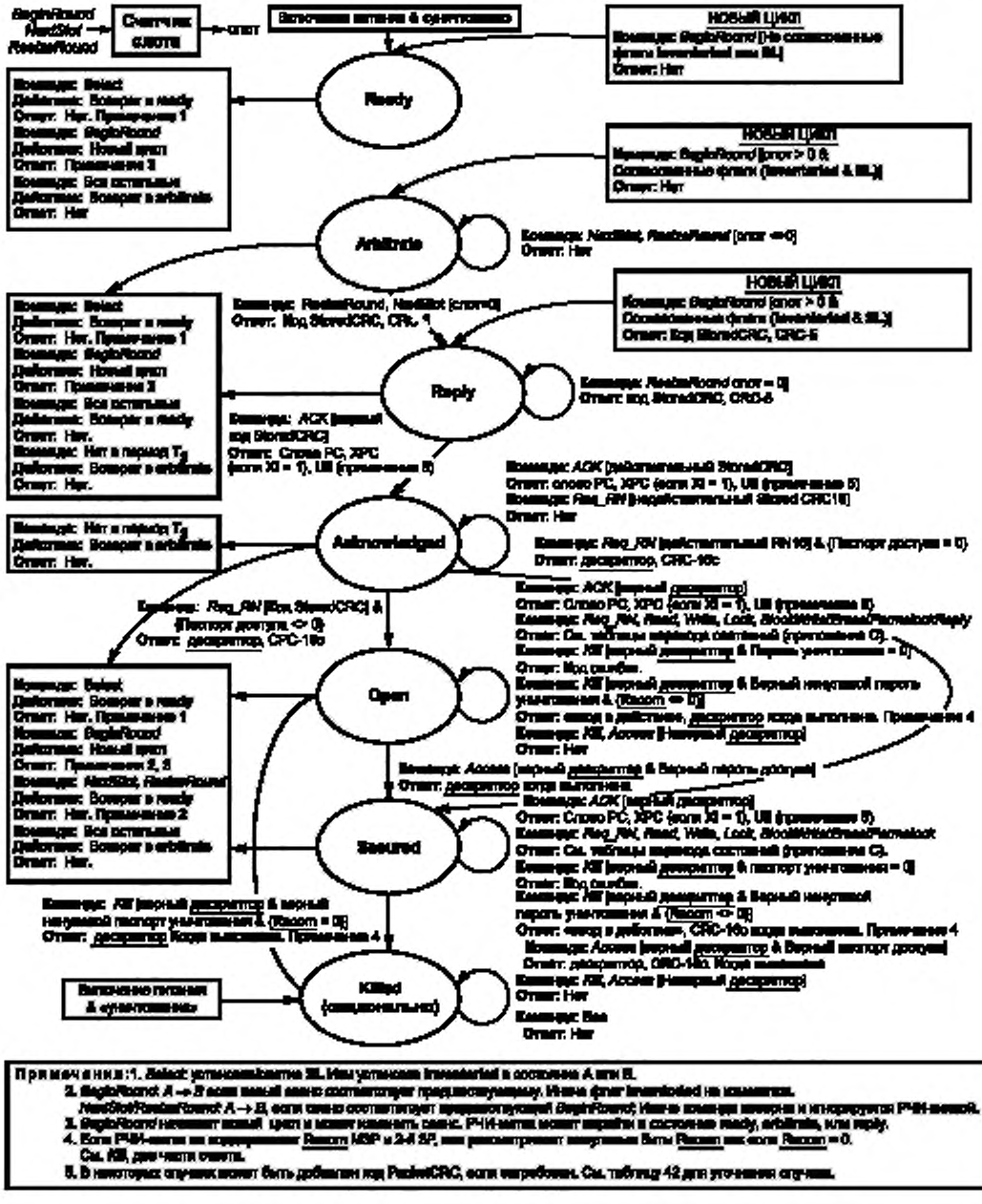



Рисунок 43 — Диаграмма состояний радиочастотной метки

6.3.3.4.4.3 Состояние **reply** (**ответ**)

Радиочастотная метка должна включать в себя реализацию состояния **reply**. После перехода в состояние **reply** радиочастотная метка должна передать код StoredCRC. Если радиочастотная метка принимает действительное подтверждение (ACK), то она должна перейти в состояние **acknowledged**, передав свои значения слов PC, XPC (если активирован XI), UII и, если затребован, код PacketCRC (см. таблица 42). Если радиочастотной метке не удалось принять команду *ACK*, или была принята не-

верная команда *ACK*, она должна вернуться в состояние **arbitrate**. Радиочастотная метка и УСО должны соответствовать всем требованиям, указанным в таблице 41.

6.3.3.4.4.4 Состояние **acknowledged** (**подтверждена**)

Радиочастотная метка должна включать в себя реализацию состояния **acknowledged**. Радиочастотная метка в состоянии **acknowledged** в зависимости от принятой команды может перейти в любое другое состояние за исключением состояния **killed** (см. рисунок 43). Если радиочастотная метка в состоянии **acknowledged** примет действительную команду *ACK*, содержащую правильные коды StoredCRC, то она должна повторить передачу, ответив, как показано в таблице 42. Если радиочастотной метке в состоянии **acknowledged** не удается принять действительную команду за интервал $T_{2(max)}$, она должна вернуться в состояние **arbitrate**. Радиочастотная метка и УСО должны соответствовать всем требованиям, указанным в таблице 41.

6.3.3.4.4.5 Состояние **open** (**открыта**)

Радиочастотная метка должна включать в себя реализацию состояния **open**. Радиочастотная метка в состоянии **acknowledged** при ненулевом пароле *доступа* должна перейти в состояние **open** после приема команды *Req_RN*, передать свой RN16 (обозначаемый как маркер подлинности), который УСО должно использовать в последующих командах, и который радиочастотная метка должна использовать в последующих ответах. Радиочастотные метки в состоянии **open** могут выполнять все команды, за исключением команд *Lock* и *BlockPermalock*. Радиочастотная метка в состоянии **open** в зависимости от принятой команды может перейти в любое другое состояние за исключением состояния **acknowledged** (см. рисунок 43). Если радиочастотная метка в состоянии **open** примет действительную команду *ACK*, содержащую правильный маркер подлинности, то она должна повторить передачу, ответив, как показано в таблице 42. Радиочастотная метка и УСО должны соответствовать всем требованиям, указанным в таблице 41, кроме значения $T_{2(max)}$; в состоянии **open** максимальная временная задержка между ответом радиочастотной метки и передачей УСО не ограничена.

6.3.3.4.4.6 Состояние **secured** (**закрыта**)

Радиочастотная метка должна включать в себя реализацию состояния **secured**. Радиочастотная метка в состоянии **acknowledged** при ненулевом пароле *доступа* должна перейти в состояние **secured** после приема команды *Req_RN*, передать свой RN16 (обозначаемый как маркер подлинности), который УСО должно использовать в последующих командах, а радиочастотная метка должна использовать в последующих ответах. Радиочастотная метка в состоянии **open** при ненулевом пароле *доступа* должна перейти в состояние **secured** после приема действительной команды *Access*, содержащей тот же маркер подлинности, который предварительно был передан при переходе из состояния **acknowledged** в состояние **open**. Радиочастотная метка в состоянии **secured** может выполнять все команды *доступа*. Радиочастотная метка в состоянии **secured** в зависимости от принятой команды может перейти в любое другое состояние за исключением состояния **open** или состояния **acknowledged** (см. рисунок 43). Если радиочастотная метка в состоянии **secured** примет действительную команду *ACK*, содержащую правильный маркер подлинности, то она должна повторить передачу, ответив, как показано в таблице 42. Радиочастотная метка и УСО должны соответствовать всем требованиям, указанным в таблице 41, кроме значения $T_{2(max)}$; в состоянии **secured** максимальная временная задержка между ответом радиочастотной метки и передачей УСО не ограничена.

6.3.3.4.4.7 Состояние **killed** (**уничтожена**) (необязательное)

Радиочастотная метка должна включать в себя реализацию состояния **killed**. Если состояние **killed** поддерживается, то радиочастотная метка, находящаяся либо в состоянии **open**, либо в состоянии **secured** должна переходить в состояние **killed** после приема действительной команды *Kill* (см. 6.3.3.4.11.3.4) при ненулевом действительном пароле *уничтожения*, нулевых значениях битов Recom команды (см. 6.3.3.4.10) и действительном маркере подлинности. Если радиочастотная метка не поддерживает работу с младшим и вторым младшим значащим битом Recom, то она обрабатывает ненулевые биты Recom как будто Recom = 0. Команда ***Kill*** полностью блокирует радиочастотную метку. После перехода в состояние **killed** радиочастотная метка должна уведомить УСО, что команда ***Kill*** была выполнена успешно, и впоследствии не должна отвечать УСО. Радиочастотные метки должны находиться в состоянии **killed** при всех условиях, и должны немедленно переходить в состояние **killed** при последующем включении питания. Состояние **killed** радиочастотной метки необратимо.

6.3.3.4.4.8 Счетчик слота

Радиочастотные метки должны включать в себя реализацию 15-битового счетчика слота. После приема команды *BeginRound* или *ResizeRound* радиочастотная метка должна предварительно загрузить значение между 0 и $2^Q$–1 из своего RNG (см. 6.3.3.4.5) в свой счетчик слота. *Q* — целое в диа-

пазоне (0, 15). Команда *BeginRound* определяет *Q*; команда *ResizeRound* может изменить *Q* из предшествующей команды *BeginRound*. После приема команды *NextSlot* радиочастотная метка должна декрементировать свой счетчик слота. Счетчик слота должен быть способен к непрерывному отсчету: подразумевается, что после декрементирования счетчика слота до $0000_h$ он должен переполниться и начать новый отсчет в режиме вычитания со значения $7FFF_h$. См. также приложение F.

Радиочастотные метки должны генерировать 16-битовые случайные или псевдослучайные числа (RN16), используя генератор случайных чисел (RNG), и должны иметь возможность извлекать *Q*-битовые подгруппы из числа RN16 для предварительной загрузки счетчика слота радиочастотной метки.

Радиочастотные метки в состоянии **arbitrate** декрементируют свой счетчик слота каждый раз, когда принимают команду *NextSlot*, при совпадающем параметре сеанс, когда их счетчик слота достигает **$0000_h$**, они переходят в состояние **reply** и передают в ответ *код StoredCRC*. Радиочастотные метки, счетчик слота которых достиг $0000_h$, отправившие ответ и не получившие подтверждения* (включая те метки, которые сразу ответили на первоначальную команду *BeginRound* и не получившие подтверждения**), должны вернуться в состояние **arbitrate** со значением слота $0000_h$ и по следующей команде *NextSlot* декрементировать это значение с $0000_h$ на $7FFF_h$. Счетчик слота должен быть способен к непрерывному отсчету. Подразумевается, что после перехода к значению $7FFF_h$ он начнет считывать снова. Таким образом эффективно предотвращаются последующие ответы до тех пор, пока радиочастотная метка не загрузит новое случайное значение в свой счетчик слота. См. также приложение F.

Примечание — В цикле инвентаризации УСО дает команду радиочастотной метке на загрузку *Q*-битового случайного числа (или псевдослучайного) в ее счетчик слота; УСО также может дать команду радиочастотной метке на декрементирование ее счетчика слота. Радиочастотные метки отвечают, когда значение в их счетчике слота (т. е., их слот — см. ниже) нулевое. *Q* — целое число в диапазоне (0, 15); соответственно вероятность получения подтверждающего ответа радиочастотной меткой лежит в диапазоне от $2^0 = 1$ до $2^{-15} = 0{,}000031$.

6.3.3.4.5 Методы ASK и PJM: Генератор случайных или псевдослучайных чисел радиочастотной метки

Радиочастотные метки должны включать в себя реализацию генератора случайных или псевдослучайных чисел (RNG). RNG должен удовлетворять следующему критерию случайности, независимо от интенсивности возбужденного поля, скорости соединения R=>T и скорости сохранения данных в радиочастотной метке (включая слово *PC*, слово *XPC*, *UII* и CRC-16). RNG используется для выработки псевдослучайных чисел RN16, маркера подлинности и счетчика слота. Радиочастотные метки, поддерживающие защиту кодированием, должны иметь возможность во включенном состоянии временно хранить последние два RN16 для использования, например, для формирования маркера подлинности и реализации 16-битовую защиту кодированием во время передачи пароля (см. рисунок 47 или рисунок 49).

- **Вероятность получения заданного значения RN16:** Вероятность извлечения любого кода RN16 из RNG, имеющего значение RN16 = *j*, для любого *j*, должна быть ограничена при $0{,}8/2^{16} < P(RN16 = j) < 1.25/2^{16}$.

- **Вероятность одновременных идентичных последовательностей:** Для множества радиочастотных меток до десяти тысяч меток, вероятность того, что любые две или более радиочастотные метки одновременно сгенерируют одну и ту же последовательность кодов RN16, должна быть менее 0,1 % независимо от того, когда радиочастотная метка активирована.

- **Вероятность предсказания RN16:** код RN16, извлекаемый из RNG через 10 мс после окончания $T_r$, как показано на рисунке 19, должен быть непредсказуем с вероятностью выше 0,025 %, если известно предыдущее извлеченное из RNG значение, выполненное в идентичных условиях.

6.3.3.4.6 Методы ASK и PJM: Управление множеством радиочастотных меток

УСО управляют множествами радиочастотных меток, используя три основные операции, показанные на рисунке 44. Каждая из этих операций включает в себя одну или несколько команд. Операции определены следующим образом:

**а) Выбор (Методы ASK и PJM):** Процесс, при котором УСО выбирает множество радиочастотных меток для инвентаризации и доступа. УСО могут использовать одну или более команд *Select* для выбора определенного множества радиочастотных меток перед инвентаризацией.

---

* *Команду ACK от УСО, переводящую ее в состояние* ***acknowledged.***

** *По причине того, что первоначальное значение RND16 сразу оказалось нулевым.*

**b) Инвентаризация в методе ASK:** Процесс, при котором УСО идентифицирует радиочастотные метки. УСО начинает инвентаризацию путем передачи команды *BeginRound* в одном из двух сеансов*. Ответить может одна или несколько меток. УСО определяет ответ одиночной радиочастотной метки и запрашивает от радиочастотной метки слово PC XPC_W1, если задано XI, и (при необходимости) необязательное слово XPC_W2, UII и код PacketCRC (см. таблицу 42). Инвентаризация производится только в одном сеансе за один раз. В Приложении Н показан пример инвентаризации и доступа УСО к одиночной радиочастотной метке.

**Инвентаризация в методе PJM:** Процесс, при котором УСО идентифицирует радиочастотные метки. УСО начинает инвентаризацию путем передачи команды *BeginRound* в одном из двух сеансов. Ответить может одна или несколько меток. УСО определяет ответ одной или более радиочастотных меток и запрашивает от радиочастотных меток слово *PC*, если задано *XI* — дополнительное слово (опционально — два слова) XPC, UII и CRC-16. Инвентаризации производится только в одном сеансе за один раз. В Приложении Н показан пример инвентаризации и доступа УСО к одиночной радиочастотной метке или множеству радиочастотных меток.

**c) Доступ в методе ASK:** Процесс, при котором УСО взаимодействует с отдельными радиочастотными метками (считывание или запись данных). Индивидуальная метка должна быть однозначно идентифицирована до начала операции доступа. Доступ включает в себя несколько команд, некоторые из которых могут использовать одноразовый шифр на основе защиты кодированием для связи R=>T.

**Доступ методом PJM:** Процесс, при котором УСО взаимодействует с отдельными радиочастотными метками или множеством радиочастотных меток (считывание или запись данных). Перед началом операции доступа радиочастотные метки должны быть однозначно идентифицированы. Доступ включает в себя несколько команд, некоторые из которых могут использовать одноразовый шифр на основе защиты кодированием для связи R=>T.

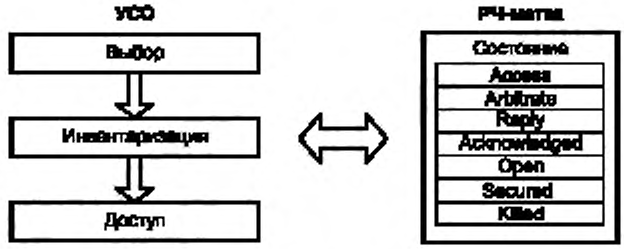



Рисунок 44 — Операции УСО/радиочастотной метки и состояния радиочастотной метки

6.3.3.4.7 Методы ASK и PJM: Выбор множества радиочастотных меток

Процесс выбора использует одиночную команду *Select*, которую УСО может последовательно применить для выбора конкретного множества радиочастотных меток на основе условия, определяемого пользователем, включающего объединение (U), пересечение (∩) и отрицание (~), основанного на выделении части множества. УСО выполняют операции U и ∩ при последовательной подаче команд *Select*. Команда *Select* может активировать или деактивировать флаг ***SL*** радиочастотной метки, либо может установить флаг **inventoried** радиочастотной метки в состояние A или B, в любом из четырех сеансов. Команда *Select* содержит параметры Target, Action, MemBank, Pointer, Length, Mask и Truncate.

- Target и Action показывают, изменяет ли команда *Select* и как изменяет флаг **SL** или флаг **inventoried** радиочастотной метки, и, в случае флага **inventoried**, для какого сеанса. Команда *Select*, которая изменяет флаг **SL**, не должна изменять флаг **inventoried**, и наоборот.

- MemBank определяет, применяется ли маска к банкам памяти UII, TID или Пользовательской памяти. *Select* применяется к одиночному банку памяти. Последующая команда *Select* может применяться к другому банку памяти.

- Pointerlength, Pointer, Length и Mask: Pointer и Length описывают диапазон памяти. Pointer ссылается на адрес бита в области памяти (Pointer не ограничен границами слова), имеющей длину 8, 16, 24 или 32 бита, как определено в Pointerlength. Длина 8 битов разрешает задавать Mask длиной от

* *См. 6.3.3.4.2.*

0 до 255 бит. Mask с большой битовой Length содержит битовую строку, которую радиочастотная метка сравнивает* с содержимым памяти, расположенным по адресам, начиная с Pointer и заканчивая последующими битами заданной Length.

- Truncate определяет, будет ли радиочастотная метка передавать полное значение своего UII, или только часть UII, расположенную непосредственно за Mask. Усечение ответа всегда влияет на значение кода StoredCRC радиочастотной метки; Радиочастотная метка не пересчитывает CRC для усеченного ответа.

При многократной выдаче одной и той же команды *Select* УСО может последовательно выбирать все множество радиочастотных меток, соответствующих критерию выбора, даже если радиочастотные метки могли подвергаться кратковременным замираниям радиочастотного сигнала.

Команда *BeginRound* использует флаги **inventoried** и **SL** для принятия решения, какие радиочастотные метки участвуют в инвентаризации. УСО имеет возможность выполнять инвентаризацию и доступ к меткам с флагами **SL** или **~SL****, либо может выбрать полное игнорирование значения флага **SL**.

6.3.3.4.8 Методы ASK и PJM: Инвентаризация множества радиочастотных меток

Набор команд инвентаризации включает в себя команды *BeginRound*, *ResizeRound*, *NextSlot*, *ACK и NAK*. Команда *BeginRound* инициирует цикл инвентаризации и выбирает, какие радиочастотные метки принимают участие в цикле (где "цикл инвентаризации" определен как интервал между следующими одна за другой командами *BeginRound*).

Команда *BeginRound* содержит параметр счетчика слота Q. После получения команды *BeginRound* принимающие участие радиочастотные метки должны выбрать случайное значение из диапазона $(0, 2^Q–1)$ включительно и должны загрузить это значение в свой счетчик слота. Радиочастотные метки, выбравшие нулевое значение, должны перейти в состояние **reply** и немедленно ответить. Радиочастотные метки, выбравшие ненулевое значение, должны перейти в состояние **arbitrate** и ожидать команду *ResizeRound* или *NextSlot*. В предположении, что отвечает одна радиочастотная метка, алгоритм выполнения ответа на команду *BeginRound* имеет следующий вид:

a) Радиочастотная метка передает код StoredCRC как перешедшая в состояние **reply**;

b) УСО дает подтверждение радиочастотной метке, подав команду *ACK*, содержащую такой же код StoredCRC;

c) Получившая подтверждение радиочастотная метка переходит в состояние **acknowledged**, дает ответ, показанный в таблице 42;

d) УСО выдает команду *ResizeRound* или *NextSlot*, означающую идентификацию радиочастотной метки, установку ее флага **inventoried** (*A*→*B*) и переход в состояние **ready**, а также подразумевающую инициирование диалога «*BeginRound*/ответ» с УСО, повторяющего все шаги описанного алгоритма сначала.

Если радиочастотная метка не может принять команду *ACK* на шаге (б) за интервал $T_2$ (см. рисунок 40 и рисунок 43), или принимает *ACK* с неправильным кодом StoredCRC, она должна вернуться в состояние **arbitrate.**

Если на шаге (а) ответит несколько РЧ меток, то УСО, определив и обработав коллизию на уровне формы сигнала, в результате получит код StoredCRC от одной из радиочастотных меток и может подать команду *ACK* этой выбранной радиочастотной метке. Невыбранные радиочастотные метки получат ошибочные коды StoredCRC и возвратятся в состояние **arbitrate** без ответа, показанного в таблице 42.

Если УСО посылает действительную команду *ACK* (т. е. *ACK*, содержащую правильный код StoredCRC для радиочастотной метки в состоянии **acknowledged**), радиочастотная метка должна послать ответ, показанный в таблице 42.

В любой момент УСО может выдать команду *NAK*, под влиянием которой все радиочастотные метки, выполняющие цикл инвентаризации, должны вернуться в состояние **arbitrate** без изменения своего флага **inventoried**.

После подачи команды *BeginRound* для инициации цикла инвентаризации УСО обычно выдает одну или более команд *ResizeRound* или *NextSlot*. Команда *ResizeRound* повторяет предыдущую команду *BeginRound* и может инкрементировать или декрементировать значение *Q*, но при этом она не вводит новые радиочастотные метки в цикл инвентаризации. *NextSlot* повторяет предыдущую команду *BeginRound* без изменения какого-либо параметра и без введения новых радиочастотных меток в цикл.

---

* *Используя логические функции И, ИЛИ.*

** *Установлен или сброшен.*

Цикл инвентаризации может содержать несколько команд *ResizeRound* или *NextSlot*. В то же время, для запуска нового цикла инвентаризации УСО должно выдать новую команду *BeginRound*.

Радиочастотные метки в состоянии **arbitrate** или **reply**, которые впервые приняли команду *ResizeRound*, модифицируют *Q* (инкрементируют, декрементируют или оставляют без изменения), а затем выбирают случайное значение из диапазона (0, $2^Q$–1) включительно и загружают это значение в свой счетчик слота. Радиочастотные метки, которые выбрали нулевое значение, должны переключиться в состояние **reply** и немедленно дать ответ*. Радиочастотные метки, которые выбрали ненулевое значение, должны переключиться в состояние **arbitrate** и ожидать команды *ResizeRound* или *NextSlot*.

Радиочастотные метки в состоянии **arbitrate** декрементируют свой счетчик слота каждый раз, когда принимают команду *NextSlot*, переходят в состояние **reply** и передают код StoredCRC, когда их счетчик слота достигает значения $0000_h$. Радиочастотные метки, счетчик слота которых достиг $0000_h$, отправившие ответ и не получившие подтверждения** (включая те радиочастотные метки, которые сразу ответили на первоначальную команду *BeginRound* и не получившие подтверждения***), должны вернуться в состояние **arbitrate** со значением слота $0000_h$ и по следующей команде *NextSlot* декрементировать это значение с $0000_h$ на $7FFF_h$. Таким образом эффективно предотвращаются последующие ответы до тех пор, пока радиочастотная метка не загрузит новое случайное значение в свой счетчик слота. Радиочастотные метки должны ответить как минимум на одну из $2^Q$–1 *NextSlot* команд.

В приложении G описан типовой алгоритм УСО для выбора *Q*.

Сценарий, изложенный выше, предполагает, что в одном сеансе работает одно УСО. Однако, как описано в 6.3.3.4.2, УСО может выполнять инвентаризацию множества радиочастотных меток в одном или двух сеансах. Кроме того, как описано в 6.3.3.4.11.2, каждая из команд *BeginRound*, *ResizeRound* и *NextSlot* содержит параметр сеанса. Ответ радиочастотной метки на эти команды изменяется в зависимости от команды, параметра сеанса и состояния метки следующим образом:

- *BeginRound*: Команда *BeginRound* начинает цикл инвентаризации и выбирает параметр сеанса для цикла. Радиочастотные метки в любом состоянии кроме **killed** должны выполнять команду *BeginRound*, начиная новый цикл в указанном сеансе и переходя в состояние **reply**, **arbitrate** или **reply** по обстоятельствам (см. рисунок 43).
  - Если радиочастотная метка в состоянии **acknowledged**, **open** или **secured** принимает команду BeginRound, чей параметр сеанса соответствует предшествующему сеансу, то она должна установить свой флаг **inventoried** (A→B)*4 для этого сеанса перед определением, будет ли она переходить в состояние **ready**, **arbitrate** или **reply**.
  - Если радиочастотная метка в состоянии **acknowledged**, **open** или **secured** принимает команду BeginRound, чей параметр сеанса не соответствует предшествующему сеансу, то она должна сохранить свой флаг **inventoried** для предшествующего сеанса без изменения, поскольку он определяет, переход в состояние **ready**, **arbitrate** или **reply**.
- *ResizeRound*, *NextSlot*: Радиочастотные метки в любом состоянии, за исключением **reply** или **killed**, должны выполнять команду *ResizeRound* или *NextSlot* если, и только если параметр сеанса в команде соответствует параметру сеанса в команде *BeginRound*, которой был запущен цикл. Радиочастотные метки должны игнорировать команду *ResizeRound* или *NextSlot* с несоответствующим параметром сеанса.
  - Если радиочастотная метка в состояниях **acknowledged**, **open** или **secured** принимает команду *ResizeRound* или *NextSlot*, чьи параметры сеанса соответствуют параметру сеанса в предшествующей команде *BeginRound*, то она должна установить свой флаг **inventoried** (A→B) для текущего сеанса, а затем перейти в состояние **ready**.

Для иллюстрации выполнения инвентаризации рассмотрим конкретный пример. Предположим, что имеется множество из 64 включенных радиочастотных меток в состоянии **ready**. УСО сначала выдает команду *Select* для выбора подгруппы радиочастотных меток. Предположим, что критерию выбора отвечают 16 радиочастотных меток. Дополнительно предположим, что 12 из 16 выбранных радиочастотных меток имеют свои флаги **inventoried**, установленные в состояние A в сеансе S0. УСО передает команду *BeginRound*, установив **(SL,** Q = 4, S0). Каждая из 12 радиочастотных меток выбирает случайное число в диапазоне (0,15) и загружает это значение в свой счетчик слота. Радиочастотные

---

* *Передают в ответ свой код StoredCRC и ожидают начало нового цикла — команду BeginRound.*

** *Команду ACK от УСО, переводящую ее в состояние* ***Acknowledged****.*

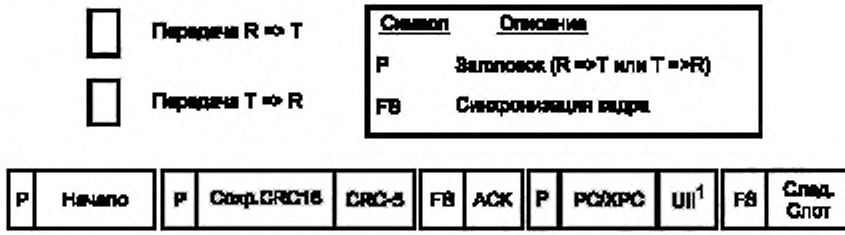
*** *По причине того, что первоначальное значение RND16 сразу оказалось нулевым.*

*4 См. 6.3.3.4.2.*

метки, выбравшие ноль, отвечают немедленно. Передача команды *BeginRound* может иметь 3 возможных результата:

a) **Нет ответа радиочастотных меток:** УСО может выдать другую команду *BeginRound*, либо может выдать команды *ResizeRound* или *NextSlot*.

b) **Ответила одна радиочастотная метка** (см. рисунок 45). Радиочастотная метка переключается в состояние **reply** и выполняет передачу кода StoredCRC. УСО подтверждает получение кода StoredCRC радиочастотной метки передачей команды *ACK*. Если радиочастотная метка принимает команду *ACK* с правильным кодом StoredCRC, она передает ответ, как показано в таблице 42, и переходит в состояние **acknowledged**. Рисунок сделан в предположении, что параметр *XI* деактивирован. Если радиочастотная метка принимает команду *ACK* с неправильным кодом StoredCRC, то она переходит в состояние **arbitrate.** Совершая успешную команду *ACK*, УСО может получить доступ к подтвержденной радиочастотной метке*, либо подать команду *ResizeRound* или *NextSlot* для установки флага **inventoried** радиочастотной метки из *A*→*B* и переводу радиочастотной метки в состояние **ready** (команда *BeginRound* с параметром сеанса, соответствующим предыдущему циклу инвентаризации, также установит флаг **inventoried** из *A*⇢*B*).




Примечание 1 — В некоторых случаях должен быть добавлен *PacketCRC*. Для уточнения см. таблицу 42.

Рисунок 45 — Ответ одной радиочастотной метки

c) **Ответ от множества радиочастотных меток:** УСО отслеживает форму сигнала, модулированного множеством кодов StoredCRC. Он может пытаться обработать коллизии и передавать команды *ACK*; не обрабатывать коллизии и передавать команды *ResizeRound*, *NextSlot* или *NAK*; или быстро идентифицировать коллизии и выдавать команды *ResizeRound* или *NextSlot* до того, как конфликтующие радиочастотные метки закончат передачу. В последнем случае конфликтующие радиочастотные метки, не получившие действительный ответ в интервале $T_2$ (см. рисунок 40), должны вернуться в состояние **arbitrate** и ожидать следующей команды *BeginRound* или *ResizeRound*.

Метод PJM: УСО могут принимать множественные ответы радиочастотных меток на множестве каналов в ответ на параметры в командах *BeginRound*, *NextSlot* или *ResizeRound*. Таким образом УСО имеет возможность принять (без какой-либо ошибки перекрытия) до восьми радиочастотных меток за одну команду *BeginRound*, *NextSlot* или *ResizeRound*. Однако, при правильно заданном *Q*, сравниваемым с неизвестным множеством радиочастотных меток, средняя оценка приема для УСО, имеющего восемь приемных каналов за одну команду *BeginRound*, *NextSlot* или *ResizeRound*, предполагается следующей:

- три канала по одной радиочастотной метке в каждом (следовательно, три радиочастотных метки принимаются правильно, например, радиочастотная метка 1 на канале A, радиочастотная метка 2 на канале B и радиочастотная метка 3 на канале E);
- один канал с двумя налагающимися радиочастотными метками;
- один канал с тремя налагающимися радиочастотными метками;
- три пустых канала.

Выбор канала определяется кодом StoredCRC (см. подраздел 6.3.3.3.1.3.11).

Если три радиочастотных метки правильно принимаются, УСО может подтвердить всем трем радиочастотным меткам подачей одной команды *ACK-PJM*, см. 6.3.3.4.11.2.4. В текущем примере команда *ACK-PJM* содержит три *кода StoredCRC* радиочастотных меток, которые были приняты. По завершении команды ACK-PJM, радиочастотная метка получает корректный код StoredCRC, который она использует для передачи своих *PC, UII* и, при необходимости, кода PacketCRC (см. таблицу 42), используя тот же

* *Переведенной в состояние* ***acknowledged*** *командой ACK данного УСО.*

канал связи, как в ответах на предшествующие команды *BeginRound*, *NextSlot* или *ResizeRound*. Затем радиочастотная метка переходит в состояние **acknowledged**. В этом случае три радиочастотных метки одновременно отвечают своими PC, UII и, при необходимости, кодом PacketCRC (см. таблицу 42) на отдельных каналах каждая (радиочастотная метка 1 на канале A, радиочастотная метка 2 на канале B и радиочастотная метка 3 на канале E). При необходимости УСО может связать коды StoredCRC радиочастотных меток с кодами UII, так как каждая радиочастотная метка использует один и тот же канал для ответов на команды *BeginRound*, *NextSlot* или *ResizeRound* и последующую *ACK-PJM*.

Если радиочастотная метка принимает команду *ACK-PJM*, в которой все коды StoredCRC неверны, она переходит в состояние **arbitrate**.

Совершая успешные команды *ACK-PJM*, УСО может получить доступ к подтвержденным радиочастотным меткам, либо подать команду *NextSlot* или *ResizeRound* для установки флагов **inventoried** радиочастотных меток из A→B и переводу их в состояние **ready** (команда *BeginRound* с параметром сеанс, соответствующим предыдущему циклу, также устанавливает флаг **inventoried** из A→B).

Радиочастотные метки, о которых шла речь выше, являются типовыми и предназначенными по своему устройству для приема команд PJM.

6.3.3.4.9 Методы ASK и PJM: Работа с отдельными радиочастотными метками

После подтверждения радиочастотной метки УСО может получить к ней доступ. Набор команд для доступа включает в себя команды *Req_RN*, *Read*, *Write*, *Kill*, *Lock*, *Access*, *BlockWrite*, *BlockErase* и *BlockPermalock*. Радиочастотные метки выполняют команду *Req_RN* в состояниях **acknowledged**, **open** или **secured**. Радиочастотные метки выполняют команды *Read*, *Write*, *Kill*, *Lock*, *Access*, *BlockWrite u BlockErase* в состояниях **open** или **secured**. Радиочастотные метки выполняют команду *Lock* только в состоянии **secured**.

УСО получает доступ к радиочастотной метке в состоянии **acknowledged** следующим образом:

Шаг 1. УСО выдает команду *Req_RN* для подтверждения радиочастотной метки.

Шаг 2. Радиочастотная метка генерирует и сохраняет RN16 (обозначаемый как маркер подлинности), передает свой маркер подлинности и переходит в состояние **open**, если ее пароль доступа ненулевой, либо переходит в состояние **secured**, если ее пароль доступа нулевой. После чего УСО может передавать следующие команды доступа.

Все команды доступа, переданные радиочастотной метке в состояниях **open** или **secured,** содержат маркер подлинности как параметр команды. Находясь в одном из этих состояний, радиочастотные метки перед выполнением команд доступа должны убедиться, что маркер подлинности правильный, и должны игнорировать команды доступа при неправильном маркере подлинности. Значение маркера подлинности неизменно на все время действия последовательности доступа.

Радиочастотные метки в состоянии **open** могут выполнять все команды, за исключением команд *Lock* и *BlockPermalock*. Радиочастотная метка в состоянии **secured** может выполнять все команды доступа. Ответ радиочастотной метки на команду доступа включает, как минимум, ее маркер подлинности; ответ может также включать другую информацию (например, результат операции *Read)*.

УСО может передать команду *ACK* радиочастотной метке в состояниях **open** или **secured**, заставляющую радиочастотную метку выполнить передачу ответа, показанного в таблице 42.

УСО и радиочастотная метка могут неограниченно обмениваться информацией в состояниях **open** или **secured**. УСО может завершить связь в любое время, подав команду *BeginRound*, *ResizeRound*, *NextSlot* или *NAK*. Ответ радиочастотной метки на команды *BeginRound*, *ResizeRound* или *NextSlot* описан в 6.3.3.4.8. *NAK* заставляет все радиочастотные метки, находящиеся в цикле инвентаризации, вернуться в состояние **arbitrate** без изменения их флага **inventoried**.

Команды *Write*, *Kill* и *Access* передаются 16-битовыми словами (данные или половинки пароля) от УСО в радиочастотную метку. Опционально эти команды могут использовать одноразовый шифр защиты кодированием соединения для сокрытия данных, передавая слова следующим образом:

Шаг 1. УСО передает команду *Req_RN*, на которую радиочастотная метка отвечает передачей нового кода RN16. Затем УСО генерирует 16-битовую строку символов защиты кодированием, полученную путем поразрядной операции XOR (ИСКЛЮЧАЮЩЕЕ ИЛИ) 16-битовых слов для передачи вместе с новым кодом RN16, и то и другое старшим значащим битом вперед, и передает команду, содержащую эту строку зашифрованных символов как параметр.

Шаг 2. Радиочастотная метка дешифрирует принятую строку символов защитой кодированием путем выполнения поразрядной операции XOR 16-битовых символов принятой зашифрованной строки с оригинальным RN16.

УСО не использует маркер подлинности для защиты кодированием.

УСО не должно повторно использовать код RN16 для защиты кодированием, если он был использован. Если УСО повторно передает команду, которая содержит данные защиты кодированием, то УСО должно повторно передать команду без изменений. Если УСО изменило данные и использует защиту кодированием, то сначала оно должно передать команду *Req_RN* для получения нового кода RN16, а затем использовать этот новый код RN16 для защиты кодированием.

Если не был запрошен код RN16, следующее 16-битовые слова (данные или половинки пароля) должны передаваться открытым текстом без защиты кодированием. УСО должно в начале передачи последовательности доступа решать, будет использована защита кодированием или нет. Если радиочастотная метка не поддерживает защиту кодированием, она должна давать ответ с кодом $0000_h$ на каждую команду *Req_RN*, которая запрашивает код RN16 для защиты кодированием.

Таблица 45 — Команды доступа и состояния радиочастотных меток, в которых допускается их выполнение

| Команда | Состояние | | | Замечание |
|---|---|---|---|---|
| | подтверждение | открытое | защищено | |
| *Req_RN* | Допустимо | Допустимо | Допустимо | — |
| *Read* | — | Допустимо | Допустимо | — |
| *Write* | — | Допустимо | Допустимо | Дополнительное перед *Req_RN* |
| *Kill* | — | Допустимо | Допустимо | Дополнительное перед *Req_RN* |
| *Lock* | — | — | Допустимо | |
| *Access* | — | Допустимо | Допустимо | Дополнительная команда: дополнительное перед *Req_RN* |
| *BlockWrite* | — | Допустимо | Допустимо | Дополнительная команда |
| *BlockErase* | — | Допустимо | Допустимо | Дополнительная команда |
| *BlockPermalock* | — | — | Допустимо | Дополнительная команда |

В команде *BlockWrite* (см. таблицу 75) передается множество 16-битовых слов от УСО в радиочастотную метку и, в отличие от *Write*, *BlockWrite* не используется соединение с защитой кодированием.

Радиочастотная метка отвечает на команду, которая записывает или стирает из памяти (т. е. *Write*, *Kill*, *Lock*, *BlockWrite*, *BlockErase* и *BlockPermalock* с параметром Read/Lock = 1 (см. п. 6.3.3.4.11.3.9)) передачей ее маркера подлинности, показывая, что операция была успешно завершена, либо передачей кода ошибки (см. приложение Е), показывая, что операция не была успешно завершена. В ответе радиочастотной метки используется один из заголовков, показанных на рисунке 27, на рисунке 31 и на рисунке 35, по обстоятельствам. См. в 6.3.3.4.11.3 подробное описание ответа радиочастотной метки на каждую отдельную команду доступа.

Передача *пароля* доступа на радиочастотную метку — это многошаговая процедура, описанная в 6.3.3.4.11.3.6 и показанная на рисунке 49.

Память радиочастотной метки может быть разблокирована или заблокирована. Состояние блокировки может быть изменяемым или постоянным (т. е., «постоянно неблокированным» или «постоянно заблокированным»). «Ввод в действие» радиочастотной метки может изменить состояние блокировки, даже если память была предварительно «постоянно заблокирована». УСО может производить запись в неблокированную память из любого состояния: **open** или **secured**. УСО может записывать в заблокированную память, которая не «постоянно заблокирована», только из состояния **secured**. См. 6.3.3.4.10, 6.3.3.4.11.3.5, 6.3.3.4.11.3.9, таблицу 70 и таблицу 80, чтобы получить подробное описание блокировки памяти, «постоянной блокировки», «ввода в действие» и состояний радиочастотной метки, требующихся для модификации памяти.

«Уничтожение» радиочастотной метки — это многошаговая процедура, описанная в 6.3.3.4.11.3.4 и указанная в рисунке 47.

Передача пароля доступа к радиочастотной метке — это многошаговая процедура, описанная в 6.3.3.4.11.3.6 и показанная на рисунке 49.

УСО и радиочастотная метка должны передавать все строки данных, начиная со старшего значащего бита.

УСО не должны выключать питание, пока радиочастотная метка находится в состояниях **reply**, **acknowledged**, **open** или **secured**. УСО должно заканчивать свой диалог с радиочастотной меткой до выключения питания, сохраняя радиочастотную метку либо в состоянии **ready**, либо в состоянии **arbitrate**.

Метод PJM: УСО могут обмениваться информацией и получать ответы одновременно от нескольких радиочастотных меток.

Примечание — Это характерно для многоканального метода PJM. Для команд *Kill* и *Access* УСО должно связываться только с одной радиочастотной меткой за один раз. Однако, для команд *Req_RN*, *Read*, *Write*, *Lock*, *BlockWrite*, *BlockErase* и *BlockPermalock* УСО может расширить поле RN для включения многих кодов StoredCRC или маркеров подлинности, см. 6.3.3.4.11, так что УСО может связываться одновременно с многими радиочастотными метками. Радиочастотная метка должна отвечать на такие команды, используя тот же канал, который был использован ею для ответа на предшествующую команду *BeginRound*, *NextSlot* или *ResizeRound*.

6.3.3.4.10 «Уничтожение» (опционально) или «ввод в действие» радиочастотной метки

«Уничтожение» или «ввод в действие» радиочастотной метки – это многошаговая процедура, описанная в 6.3.3.4.11.3.4 и показанная на рисунке 47, в которой УСО передает радиочастотной метке поочередно две команды *Kill*. Первая команда *Kill* содержит первую часть пароля уничтожения, а вторая команда *Kill* содержит вторую часть. Каждая команда *Kill* также содержит 3 бита RFU/Recom. В первой команде *Kill* это биты RFU*, и они имеют нулевое значение; во второй команде *Kill* они содержат биты «ввода в действие» (или Recom) и могут иметь ненулевое значение. Процедуры для «уничтожения» или «ввода в действие» радиочастотной метки идентичны, за исключением того, что при «уничтожении» биты «ввода в действие» во второй команде *Kill* имеют нулевые значения и при ее «вводе в действие» имеют ненулевые значения. Независимо от предполагаемой операции радиочастотная метка не должна «уничтожаться» или повторно «вводиться в действие» самостоятельно, без принятого сначала правильного пароля уничтожения согласно процедуре, как показано на рисунке 47.

Если в радиочастотной метке не реализованы средства обработки наименьшего значащего бита и второго младшего бита Recom, то она должна их игнорировать и обрабатывать их как нулевые. Если радиочастотная метка принимает правильно форматированную команду *Kill* с правильным паролем уничтожения и тремя битами Recom, установленными в ноль, а эта команда не поддерживается, так как она опциональная, она должна быть интерпретирована, как будто 3 младших значащих бита Recom установлены в единицу. Далее в данном пункте 6.3.3.4.10 предполагается, что радиочастотная метка выполняет «ввод в действие» с поддержкой наименьшего значащего бита и второго младшего значащего бита Recom.

После приема правильно форматированной команды *Kill* с правильным паролем уничтожения и одним или более ненулевым битом «ввода в действие», радиочастотная метка должна установить младшие значащие биты своего слова *XPC_W1*, как они установлены в битах «ввода в действие» (например, если радиочастотная метка принимает биты «ввода в действие» $100_2$, то она устанавливает третий младший значащий бит своего слова XPC_W1). Для младших значащих битов слова XPC_W1 должна быть реализована возможность однократной записи, это означает, что они не могут быть сняты после их установки. Сохранение состояния «ввода в действие» радиочастотной метки в слове XPC_W1 дает впоследствии возможность УСО узнать, каким образом радиочастотная метка была «введена в действие» (см. таблицу 43). Радиочастотная метка должна выполнять следующие операции, основанные на установленных ей значениях битов «ввода в действие» (см. также 6.3.3.4.1.2.5):

- **Установлен младший значащий бит:** радиочастотная метка должна отменить «постоянную блокировку» блоков и разблокировать все блоки Пользовательской памяти, которые были ранее «постоянно заблокированы». Радиочастотная метка должна отменить поддержку команды *BlockPermalock*. Если у радиочастотной метки перед «вводом в действие» не реализована «постоянная блокировка» блоков, то «постоянная блокировка» и должна оставаться отключенной. Состояние блокировки Пользовательской памяти должно определяться исключительно битами блокировки (см. п. 6.3.3.4.11.3.5).

- **Установлен второй младший значащий бит:** Радиочастотная метка должна сделать свою Пользовательскую память недоступной, делая тем самым весь банк памяти нечитаемым, незаписываемым и невыбираемым (т. е. радиочастотная метка работает так, как будто ее банк Пользовательской

* *Статус — «для будущего использования».*

памяти больше не существует). Второй младший значащий бит имеет более высокий приоритет, чем младший значащий бит — если установлены оба, то память пользователя недоступна.

- **Установлен третий младший значащий бит:** радиочастотная метка должна разблокировать свои банки памяти UII, TID и банк Пользовательской памяти, независимо от того, были ли эти банки заблокированы или «постоянно заблокированы». Участки Пользовательской памяти, которые имели «постоянную блокировку» блока, должны ее сохранить и наоборот, но если младший значащий бит *XPC* тоже активирован, то радиочастотная метка должна разблокировать все блоки, имеющие «постоянную блокировку». Радиочастотная метка должна разблокировать для записи свои пароли уничтожения и доступа. Статус Read/Lock паролей уничтожения и доступа должен оставаться в том же состоянии, как перед «вводом в действие». Если пароль уничтожения или доступа был нечитаемым перед «вводом в действие», и в дальнейшем УСО попытается их считывать, радиочастотная метка должна выполнить передачу кода ошибки (см. приложение Е). Радиочастотная метка, принявшая команду *Lock* с нулевым паролем считывания/записи должна заблокировать или «постоянно заблокировать» для записи указанный пароль (пароли).

Радиочастотная метка не должна выполнять рассмотренные выше операции «ввода в действие» более одного раза. Как один из примеров, никакие банки памяти радиочастотной метки не могут быть разблокированными более одного раза при «вводе в действие».

Радиочастотная метка может выполнить несколько последовательных команд *Kill*, но с ограничениями в зависимости от смысла и назначения операций, указанных в этих командах. Конкретно:

- «Уничтоженная» радиочастотная метка не должна позволять последующий «ввод в действие».

- Ранее «введенная в действие» радиочастотная метка, которая приняла должным образом отформатированную команду *Kill* с правильным *паролем уничтожения* и Recom = $000_2$, должна быть «уничтожена».

- Радиочастотная метка, которая приняла должным образом отформатированную команду *Kill* с правильным паролем уничтожения, но с битами «ввода в действие», уже зарезервированными в слове XPC_W1 (например, биты «ввода в действие» имеют значение $100_2$, но слово XPC_W1 радиочастотной метки уже содержит значение $100_2$), не должна снова выполнять требуемую операцию «ввода в действие». Вместо этого радиочастотная метка должна только проверить, что слово XPC_W1 содержит установленные значения, и подтвердить УСО выполнение команды. УСО может отправить команду *Kill* с зарезервированными битами «ввода в действие», например, в случае, если перед этим была отправлена команда *Kill*, но не было получено подтверждения о ее выполнении от радиочастотной метки.

- Радиочастотная метка, которая приняла должным образом отформатированную команду *Kill* с правильным паролем уничтожения и по-иному установленными битами «ввода в действие», должна выполнить операцию «ввода в действие», указанную по-новому установленными битами, и ответить подтверждением выполнения УСО. Радиочастотная метка должна произвести операцию логического ИЛИ младших значащих битов ее текущего XPC_W1 и битов «ввода в действие» и сохранить результат в своем слове XPC_W1. Например, если радиочастотная метка с младшими значащими битами слова XPC_W1 = $100_2$ принимает команду *Kill*, биты «ввода в действие» которой имеют значение $0102_2$, то метка должна сделать свой банк Пользовательской памяти недоступным и сохранить в своем слове XPC_W1 значение $1102_2$.

УСО может впоследствии повторно заблокировать любые банки памяти или пароли, которые были разблокированы при «вводе в действие».

Если отдельные сегменты или целые банки памяти радиочастотной метки были заблокированы при производстве, они не могут быть разблокированы при вводе в строй. УСО может определить, поддерживает ли радиочастотная метка работу с наименьшим значащим битом и вторым младшим битом Recom, и если поддерживает, то какие (если есть) сегменты памяти не должны «вводиться в действие», прочитав память TID радиочастотной метки перед «вводом в действие».

Радиочастотная метка, в которой не реализован пароль уничтожения, или радиочастотная метка с нулевым паролем уничтожения не должны выполнять операции «уничтожения» или «ввода в действие». Такая радиочастотная метка должна отвечать кодом ошибки (как показано на рисунке 47) на команду *Kill* не зависимо от установок битов RFU или «ввода в действие».

Радиочастотная метка должна принимать все восемь возможных комбинаций трех битов «ввода в действие», если поддерживаются наименьший значащий бит и второй младший бит Recom, выполняя те операции в той части, которые можно выполнить, игнорируя те, которые невозможны, и отвечая УСО подтверждением выполнения по завершении. Некоторые примеры операций, которые радиочастотная метка может быть неспособной или частично способной выполнить:

- Радиочастотная метка, не имея пользовательской памяти, не может ее разблокировать;
- Радиочастотная метка с нереализованным паролем доступа не может разблокировать его для записи;
- Радиочастотная метка, у которой часть памяти TID заблокирована при производстве, не может разблокировать эту часть.

6.3.3.4.11 Методы ASK и PJM: Команды УСО и ответы радиочастотной метки

Команды УСО-радиочастотная метка должны иметь формат, показанный в таблице 46.

- Команда *ACK* имеет 2-битовый код команды $01_2$,
- Команды *BeginRound*, *NextSlot*, *ResizeRound* и *Select* имеют 4-битовые коды команд (см. таблица 46),
- Все остальные базовые команды имеют 8-битовые коды команд, начинающиеся с $110_2$,
- Все расширенные команды имеют 16-битовые коды команд, начинающиеся с $1110_2$.
- Команды *NextSlot, ACK, BeginRound, ResizeRound* и *NAK* имеют уникальные длины кода команд, показанные в таблице 46. Ни одна из других команд не должна иметь таких длин кода. Если радиочастотная метка принимает одну из этих команд с некорректной длиной кода, она должна ее игнорировать.
- Команды *BeginRound*, *ResizeRound* и *NextSlot* содержат параметр сеанса.
- Команда *BeginRound* защищена при помощи CRC-5, как показано в таблице 49, а подробно описано в приложении I.
- Команды *Select, Req_RN, Read, Write, Kill, Lock, Access, BlockWrite, BlockErase* и *BlockPermalock* защищены с помощью CRC-16c, определенного в п. 6.3.3.4.1.4, и подробно описаны в приложении I.
- Команды R=>T начинаются либо с заголовка, либо с кадра синхронизации, как описано в 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8. Длина кода команд, указанная в таблице 46, не включает заголовок или кадр синхронизации.
- Радиочастотные метки должны игнорировать недействительные команды. В общем случае "недействительная" означает команду, которая (1) неправильно дает текущее состояние радиочастотной метки, (2) не поддерживается радиочастотной меткой, (3) имеет некорректные параметры, (4) имеет ошибочный CRC, (5) указывает неправильный параметр сеанс, или (6) в любом другом варианте не признается или не выполняется радиочастотной меткой. Фактическое определение термина "недействительная" зависит от специфики состояния радиочастотной метки и определяется для каждого состояния в приложении С и приложении Е.

Таблица 46 — Команды

| Команда | Код | Длина, бит | Обязательность | Защита |
|---|---|---|---|---|
| *ACK (метод ASK)* | 01 | 18 | Да | Длина кода уникальной команды |
| *ACK (метод PJM)* | 01 | >/ = 23 | Да | CRC-5 |
| *NextSlot* | 0000 | 6 | Да | Длина кода уникальной команды |
| *BeginRound* | 1000 | 22 | Да | Длина кода уникальной команды и CRC-5 |
| *ResizeRound* | 1001 | 9 | Да | Длина кода уникальной команды |
| *Select* | 1010 | >46 | Да | CRC-16c |
| Зарезервировано для будущего использования | 1011 | — | — | — |
| *NAK* | 11000000 | 8 | Да | Длина кода уникальной команды |
| *Req_RN* | 11000001 | 40 | Да | CRC-16c |
| *Read* | 11000010 | >59 | Да | CRC-16c |
| *Write* | 11000011 | >60 | Да | CRC-16c |
| *Kill* | 11000100 | 59 | Да | CRC-16c |
| *Lock* | 11000101 | 60 | Да | CRC-16c |

*Окончание таблицы 46*

| Команда | Код | Длина, бит | Обязательность | Защита |
|---|---|---|---|---|
| *Access* | 11000110 | 56 | Нет | CRC-16c |
| *BlockWrite* | 11000111 | >59 | Нет | CRC-16c |
| *BlockErase* | 11001000 | >59 | Нет | CRC-16c |
| *BlockPermalock* | 11001001 | >68 | Нет | CRC-16c |
| Зарезервировано для будущего использования | 11001010 ... 11011111 | — | — | — |
| Зарезервировано для команд пользователя | 11100000 00000000 ... 11100000 11111111 | — | — | Определено изготовителем |
| Зарезервировано для команд изготовителя | 11100001 00000000 ... 11100001 11111111 | — | — | Определено изготовителем |
| Зарезервировано для будущего использования | 11100010 00000000 ... 11101111 11111111 | — | — | — |

Примечание — Для команды *Kill* обязательна поддержка установленного только третьего младшего бита Recom. См. 6.3.3.4.10.

Метод PJM: Команды *ACK-PJM, Req_RN, Read, Write, Lock, BlockWrite, BlockErase и BlockPermalock* могут включать множественные *коды* StoredCRC или маркеры подлинности (внутри поля RN). Команды *Kill* и *Access* в методе PJM содержат только единственный маркер подлинности.

Поэтому для Метода PJM команды *ACK-PJM, Req_RN, Read, Write, Lock, BlockWrite, BlockErase и BlockPermalock* могут иметь длины кода, отличные от указанных в таблице 46. Длина кода этих команд, показанная в таблице, включает только одиночные *коды StoredCRC* или маркер подлинности. Эта длина (или минимальная длина) должна увеличиваться на 16 бит для каждого дополнительного кода StoredCRC или маркера подлинности.

Команды УСО R=>T начинаются с одного из флагов (заголовок или кадр синхронизации), как описано в 6.3.3.3.1.3.10. Длина кода команды, указанная в таблице 46, не включает флаг.

6.3.3.4.11.1 Команды выбора

Набор команд выбора включает одну команду: *Select*.

6.3.3.4.11.1.1 Команда *Select* (обязательная)

Команда *Select* выбирает конкретное множество радиочастотных меток по заданному пользователем условию, включающему в себя объединение (U), пересечение (∩) и отрицание (~), основанному на выделении части множества. УСО выполняют операции U и ∩, подавая последовательно команды *Select*. Команда *Select* может установить или сбросить флаг **SL** радиочастотной метки, который применялся во всех четырех сеансах*, либо может установить флаг **inventoried** радиочастотной метки в состояние A или B в одном из четырех сеансов.

УСО и радиочастотные метки должны включать в себя средства реализации команды *Select*, приведенной в таблице 47. Параметр Target должен показывать, изменяет ли команда *Select* флаг *SL* радиочастотной метки или флаг **inventoried**, и, в случае флага **inventoried**, для какого сеанса. Действия, которые должны вызвать ответ радиочастотной метки, приведены в таблице 48. Критерии для решения,

* *См. 6.3.3.4.2.*

будет ли радиочастотная метка соответствовать или не соответствовать условиям, определены в параметрах MemBank, Pointer, Length и Mask. Параметр Truncate показывает, должен ли ответ, передаваемый меткой, быть усеченным до включения в него только битов UII и кода StoredCRC, следующих за Mask, или нет. Команда *Select* передает от УСО к радиочастотной метке следующие параметры:

- Target показывает, будет ли команда *Select* модифицировать флаг **SL** радиочастотной метки или ее флаг **inventoried**, и в случае **inventoried** вдобавок указывает один из четырех сеансов инвентаризации. Команда *Select*, которая изменяет флаг **SL**, не должна изменять флаг **inventoried**, и наоборот. Радиочастотные метки должны игнорировать команды *Select*, у которых Target имеет значения $101_2$, $110_2$, или $111_2$.

- Action показывает, будет ли у радиочастотных меток установлен или сброшен флаг **SL**, или флаг **inventoried** установлен в состояние A или B. Радиочастотные метки, подходящие под значения полей MemBank, Pointer, Length и Mask, считаются «соответствующими». Радиочастотные метки, не подходящие под содержимое этих полей, считаются «несоответствующими».

- MemBank определяет, применяется ли параметр Mask к банкам памяти UII, TID или Пользовательской памяти. Команда *Select* должна применяться только к одному банку памяти. Последующие команды *Select* могут применяться к другим банкам. Параметр MemBank не должен определять Резервную память; Если радиочастотная метка принимает команду *Select*, задающую MemBank = $00_2$, то она должна ее игнорировать. Значение MemBank = $00_2$ зарезервировано для будущего использования (RFU).

- Pointerlength, Pointer, Length и Mask: Параметры Pointer и Length описывают размер памяти. Параметр Pointer ссылается на адрес бита памяти (Pointer не ограничен границами слова) и имеет длину 8, 16, 24 или 32 бита, как определено в Pointerlength. Параметр Length длиной 8 бит разрешает Mask длиной от 0 бит до 255 бит. Маска Mask с битовой длиной Length содержит битовую строку, которую радиочастотная метка логически сравнивает с содержимым памяти, расположенным по адресам начиная с бита, заданного параметром Pointer и заканчивая последним битом, заданным параметром Length. Если Pointer и Length относятся к области памяти, которая не существует в радиочастотной метке, то радиочастотная метка должна рассматривать команду *Select* как недействительную. Если значение Length нулевое, то все радиочастотные метки должны считываться соответствующими, за исключением случая, когда значение Pointer ссылается на несуществующую в радиочастотной метке область памяти, в этом случае радиочастотная метка должна рассматривать команду *Select* как недействительную.

- Truncate: Если УСО устанавливает параметр Truncate, и если впоследствии команда *BeginRound* установит Sel=10 или Sel=11, то соответствующая радиочастотная метка должна усекать свой ответ на команду *ACK* до той части UII, которая следует сразу же за Mask, если остаток длины *UII* > 0.

УСО должны устанавливать параметр Truncate:

- в последней (и только в последней) команде Select, которую УСО передает перед отправкой команды BeginRound;

- только если команда Select имеет значение Target = $100_2$;

- только если Mask оканчивается на банке UII.

Эти ограничения не препятствуют УСО выдавать несколько команд *Select*, которые задают **SL** и/или флаги **inventoried**. Для них требуется, чтобы УСО устанавливала параметр Truncate только к последней команде *Select*, и только если эта последняя команда *Select* устанавливает флаг **SL**. Радиочастотные метки должны начинать работу при получении питания со снятым параметром Truncate.

Радиочастотные метки должны принимать решение, будет ли усекаться передаваемое ими значение UII на основании самой последней полученной команды *Select*. Если радиочастотная метка принимает команду *Select* со значением Truncate = 1, но при этом Target <> $100_2$ , радиочастотная метка должна игнорировать команду *Select*. Если радиочастотная метка принимает команду *Select*, в которой значение Truncate = 1, но при этом MemBank <> 01, радиочастотная метка должна рассматривать команду *Select* как недействительную. Если радиочастотная метка принимает команду *Select*, в которой значение Truncate = 1, MemBank = 01, но при этом маска Mask заканчивается за пределами банка UII, установленного в коде StoredPC, радиочастотная метка должна рассматривать команду Select как недействительную.

Mask может оканчиваться на последнем бите UII, в этом случае выбранная радиочастотная метка должна выполнить передачу заголовка SOF, затем $00000_2$ и EOF. Усеченные ответы никогда не содержат слов XPC_W1 или XPC_W2, потому что Mask должна оканчиваться на UII. «Введенная в действие» радиочастотная метка не должна усекать свои ответы.

«Введенная в действие» радиочастотная метка, принявшая команду *Select* с параметром Truncate = 1, должна реагировать на нее как обычно, но при ответе на последующую команду *ACK* она должна передать свои значения данных PacketPC, XPC_W1, опционально XPC_W2 (если установлен параметр XEB), UII, чья длина соответствует указанной в поле длина UII в данных StoredPC, и код PacketCRC. УСО могут использовать команду *Select* для сброса всех радиочастотных меток в сеансе в состоянии **inventoried,** статус A, подав команду *Select* с параметром Action = $000_2$ и нулевым значением Length.

Поскольку радиочастотная метка сохраняет свои данные StoredPC и данные StoredCRC в памяти UII, команда *Select* может производить их выбор.

Поскольку радиочастотная метка рассчитывает коды PacketPC и коды PacketCRC динамически и не сохраняет их в памяти, команда Select неспособна их выбирать.

Поскольку радиочастотная метка может формировать свои PC и/или CRC динамически, ее ответ на команду *Select*, чьи параметры Pointer, Length, и Mask содержат данные StoredPC или данные StoredCRC, может порождать неожиданное поведение. А именно, ответ радиочастотной метки может оказаться не «соответствующим» параметру Mask, даже если сама радиочастотная метка является «соответствующей», и наоборот. Например, предположим, что УСО выдало команду *Select*, «соответствующую» полю длины UII $\mathbf{00100_2}$ в данных StoredPC. Далее допустим, что радиочастотная метка является «соответствующей», но имеет установленный XI. Радиочастотная метка будет динамически инкрементировать свое поле длины UII до значения $00101_2$, когда отвечает на команды ACK, и будет передавать это инкрементированное значение в данных PacketPC. Радиочастотная метка будет «соответствующей», но передаваемые значения поля длины UII будут «несоответствующими».

УСО должны предворять команду *Select* кадром синхронизации (см. 6.3.3.4.3). Значение CRC-16c, которое защищает данные команды *Select*, рассчитывается относительно первого бита кода команды до бита Truncate.

Радиочастотные метки не должны отвечать на команду *Select*.

Примечание — УСО не должно передавать параметр Target равным $001_2$ и $011_2$.

Таблица 47 — Команда *Select*

| | Команда | Target | Action | MemBank | Pointer-length | Pointer | Length | Mask | Truncate | CRC-16c |
|---|---|---|---|---|---|---|---|---|---|---|
| Число битов | 4 | 3 | 3 | 2 | 2 | 8, 16, 24 или 32 | 8 | Переменная | 1 | 16 |
| Описание | 1010 | 000: **inventoried** (S0) M3P всегда 0<br>001: Недопустимо<br>010: **inventoried** (S2) M3P всегда 0<br>011: Недопустимо<br>100: **SL**<br>101: RFU<br>110: RFU<br>111: RFU | См. табл. 48 | 00: RFU<br>01: UII<br>10: TID<br>11: Пользователь | Длина указателя<br>00: 8 бит<br>01: 16 бит<br>10: 24 бит<br>11: 32 бит | Начальный адрес Маски | Длина Маски (бит) | Значение Маски | 0: запрещено усечение<br>1: Разрешено усечение | — |

Таблица 48 — Действие радиочастотной метки по значению параметра Action

| Действие | «Соответствует» | «Не соответствует» |
|---|---|---|
| 000 | Установлен **SL** или **inventoried** →*A* | Сброшен **SL** или **inventoried** →*B* |
| 001 | Установлен **SL** или **inventoried** →*A* | Ничего не делать |
| 010 | Ничего не делать | Сброшен **SL** или **inventoried** →*B* |
| 011 | Отвергнуть **SL** или (*A*→*B*, *B*→*A*) | Ничего не делать |
| 100 | Сброшен **SL** или **inventoried** →*B* | Установлен **SL** или **inventoried** →*A* |

*Окончание таблицы 48*

| Действие | «Соответствует» | «Не соответствует» |
|---|---|---|
| 101 | Сброшен **SL** или **inventoried** →*B* | Ничего не делать |
| 110 | Ничего не делать | Установлен **SL** или **inventoried** →*A* |
| 111 | Ничего не делать | Отвергнуть **SL** или (*A*→*B*, *B*→*A*) |

6.3.3.4.11.2 Команды инвентаризации

Набор команд инвентаризации включает в себя команды *BeginRound, ResizeRound, NextSlot, ACK и NAK*.

6.3.3.4.11.2.1 Команда *BeginRound* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *BeginRound*, описанной в таблице 49. Команда *BeginRound* инициирует и определяет цикл инвентаризации. Код команды *BeginRound* содержит следующие поля.

- DR (коэффициент деления), которое задает частоту соединения T=>R в соответствии с 6.3.3.3.1.3.9 и таблицей 36;
- M (циклов на символ), которое задает скорость передачи данных в битах T=>R и формат модуляции в соответствии с таблицей 37;
- Trext, которое указывает, будет ли заголовку T=>R предшествовать пилотный сигнал в соответствии с 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8;
- Sel, которое указывает, какие радиочастотные метки отвечают на команду *BeginRound* (см. 6.3.3.3.1.4.11.1.1 и 6.3.3.3.4.8);
- Session, которое указывает сеанс для цикла инвентаризации (см. 6.3.3.3.4.8);
- *Q*, которое задает число слотов в цикле инвентаризации (см. 6.3.3.3.4.8).

УСО должны передавать заголовок перед командой *BeginRound*.

Значение CRC-5 рассчитывается относительно первого бита кода команды до последнего бита *Q*. Если радиочастотная метка принимает команду *BeginRound* с ошибочным CRC-5, она должна игнорировать команду.

После приема команды *BeginRound* радиочастотные метки, «соответствующие» команде *Select* и параметру Target, выбирают случайное значение в диапазоне от 0 до ($2^Q$–1) включительно и загружают это значение в свой счетчик слота. Если радиочастотная метка при ответе на команду *BeginRound* загружает в свой счетчик слота ноль, тогда ее ответ на команду *BeginRound* должен соответствовать описанному в таблице 50; в ином случае радиочастотная метка не должна отвечать.

Команда *BeginRound* может инициировать цикл инвентаризации в новом сеансе либо в предшествующем сеансе. Если радиочастотная метка в состояниях **acknowledged**, **open** или **secured** принимает команду *BeginRound*, чей параметр сеанса соответствует предшествующему сеансу, она должна установить свой флаг **inventoried** (A →B) для этой сессии. Если радиочастотная метка в состояниях **acknowledged**, **open** или **secured** принимает команду *BeginRound*, чей параметр сеанса не соответствует предыдущему сеансу, она должна сохранить свой флаг **inventoried** для предыдущего сеанса без изменений, когда начинается новый цикл.

Радиочастотные метки должны поддерживать все параметры DR, M и Trext, указанные в таблице 36 и таблице 37.

Метод PJM: Радиочастотные метки должны интерпретировать параметры DR, M и TRext согласно таблице 38.

Радиочастотные метки в любом состоянии, кроме **killed,** должны выполнять команду *BeginRound*; радиочастотные метки в состоянии **killed** должны игнорировать команду *BeginRound*.

Таблица 49 — Команда *BeginRound*

| | Команда | DR | M | TRext | Выбор | Сеанс | RFU | Q | CRC-5 |
|---|---|---|---|---|---|---|---|---|---|
| Число битов | 4 | 1 | 2 | 1 | 2 | 2 | 1 | 4 | 5 |

*Окончание таблицы 49*

| | Команда | DR | M | TRext | Выбор | Сеанс | RFU | Q | CRC-5 |
|---|---|---|---|---|---|---|---|---|---|
| Описание | 1000 | 0: FL= 423 кГц (fc/32)<br>1: FL= 847 кГц (fc/16) | 00: ЧМ0<br>01: кодирование с помощью кода Миллера 8<br>10: Манчестерское кодирование 2<br>11: Манчестерское кодирование 4 | 0: Нет пилотного сигнала<br>1: Использовать пилотный сигнал | 00: Все<br>01: Все<br>10: ~SL<br>11: SL | 00: S0<br>01: Недопустимо<br>10: S2<br>11: Недопустимо | 0: A<br>1: RFU | От 0 до 15 | — |

Таблица 50 — Ответ радиочастотной метки на команду *BeginRound*

| Свойство | Ответ | CRC-5 |
|---|---|---|
| Число битов | 16 | 5 |
| Описание | код StoredCRC | — |

Метод PJM: Радиочастотные метки, имеющие средства реализации приема Методом PJM, должны декодировать биты DR, M и TRext согласно таблице 38.

6.3.3.4.11.2.2 Команда *ResizeRound* (обязательная)

УСО и радиочастотная метка должны включать в себя средства реализации команды *ResizeRound*, приведенной в таблице 51. Команда *ResizeRound* определяет параметр *Q* (т. е. количество слотов в цикле инвентаризации — см. 6.3.3.4.8) без изменения любых других параметров цикла.

Команда *ResizeRound* содержит следующие поля:

- Сеанс: подтверждает номер сеанса для конкретного цикла инвентаризации (см. 6.3.3.4.8 и 6.3.3.4.11.2.1). Если радиочастотная метка принимает команду *ResizeRound*, номер сеанса которой отличается от номера в команде *BeginRound*, которой был инициирован цикл, она должна игнорировать эту команду.
- UpDn: определяет, будет ли радиочастотная метка регулировать параметр *Q* и каким образом:
  - 110: Инкремент *Q* (т. е., *Q* = *Q* + 1)
  - 000: Без изменений для *Q*
  - 011: Декремент *Q* (т. е., *Q* = *Q* – 1)

Если радиочастотная метка принимает команду *ResizeRound* со значением UpDn, отличным от указанных выше, она должна игнорировать эту команду. Если радиочастотная метка со значением *Q*, равным 15, принимает команду *ResizeRound* с UpDn = 110, она должна изменить UpDn до 000 перед выполнением команды; тауже, если радиочастотная метка со значением *Q*, равным 0, принимает команду *ResizeRound* с UpDn = 011, она должна изменить UpDn до 000 перед выполнением команды.

Радиочастотные метки должны поддерживать отсчет текущего значения *Q*. Начальное значение *Q* указано в команде *BeginRound*, которая запускает цикл инвентаризации; впоследствии одна или более команд *ResizeRound* могут изменять *Q*.

Команде *ResizeRound* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.4.8).

После приема команды *ResizeRound* радиочастотные метки сначала обновляют *Q*, а затем выбирают случайное значение в диапазоне $(0, 2^{Q}-1)$ включительно и загружают это значение в свой счетчик слота. Если радиочастотная метка при ответе на команду *ResizeRound* загружает в свой счетчик слота ноль, тогда ее ответ на команду *ResizeRound* должен соответствовать описанному в таблице 52, в ином случае метка не должна отвечать. Радиочастотные метки должны отвечать на команду *ResizeRound*, только если они приняли предшествующую команду *BeginRound*.

Радиочастотные метки в состояниях **acknowledged, open** или **secured**, которые приняли команду *ResizeRound*, устанавливают свой флаг **Inventoried** (A→B) для текущего сеанса и переходят в состояние **ready**.

Таблица 51 — Команда *ResizeRound*

| Свойство | Команда | Сеанс | UpDn |
|---|---|---|---|
| Число битов | 4 | 2 | 3 |
| Описание | 1001 | 00: S0<br>01: Недопустимо<br>10: S2<br>11: Недопустимо | 110: Q = Q + 1<br>000: Без изменений для Q<br>011: Q = Q – 1 |

Таблица 52 — Ответ радиочастотной метки на команду *ResizeRound*

| Свойство | Ответ | CRC-5 |
|---|---|---|
| Число битов | 16 | 5 |
| Описание | Код StoredCRC | — |

6.3.3.4.11.2.3 Команда *NextSlot* (обязательная)

УСО и радиочастотная метка должны включать в себя средства реализации команды *NextSlot*, приведенной в таблице 53. *NextSlot* указывает радиочастотным меткам произвести декремент их счетчиков слота и, если после декремента значение счетчика становится нулевым, передать код StoredCRC + CRC-5 для УСО.

Команда *NextSlot* содержит следующие поля:

- Сеанс подтверждает номер сеанса для конкретного цикла инвентаризации (см. 6.3.3.4.8 и таблицу 53). Если радиочастотная метка принимает команду *NextSlot*, номер сеанса которой отличается от номера в команде *BeginRound*, которая инициировала цикл, она должна игнорировать эту команду.

Команде *NextSlot* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

Если радиочастотная метка при выполнении команды *NextSlot* декрементирует свой счетчик слота и декрементирует значение слота до нулевого значения, то она должна ответить на команду *NextSlot* как показано в таблице 54; в ином случае радиочастотная метка не должна отвечать. Радиочастотные метки должны выполнять команду *NextSlot*, только если они приняли предшествующую команду *BeginRound*.

Радиочастотные метки в состояниях **acknowledged**, **open** или **secured**, которые приняли команду *NextSlot*, устанавливают свой флаг **inventoried** (*A* → *B*) для текущего сеанса и переходят в состояние **ready**.

Таблица 53 — Команда *NextSlot*

| Свойство | Команда | Сеанс |
|---|---|---|
| Число битов | 4 | 2 |
| Описание | 0000 | 00: S0<br>01: Недопустимо<br>10: S2<br>11: Недопустимо |

Таблица 54 — Ответ радиочастотной метки на команду *NextSlot*

| Свойство | Ответ | CRC-5 |
|---|---|---|
| Число битов | 16 | 5 |
| Описание | Код StoredCRC | — |

6.3.3.4.11.2.4 Команда *ACK* (обязательная)

УСО и радиочастотные метки должны включать в себя реализацию команды *ACK*, приведенную в таблице 55. УСО передает команду *ACK* для подтверждения одиночной радиочастотной метки.

Команда *ACK* — повторяет код StoredCRC, если радиочастотная метка находится в состоянии **reply** или в состоянии **acknowledged**, либо маркер подлинности радиочастотной метки, если она в состоянии **open** или **secured**.

Команде *ACK* должен предшествовать кадр синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

Ответ радиочастотной метки на «успешную» команду *ACK* должен быть таким, как описано в таблице 56. Радиочастотная метка, принявшая команду *ACK* с неверным кодом StoredCRC или неверным маркером подлинности (по обстоятельствам), должна вернуться в состояние **arbitrate** без ответа, за исключением случая, когда радиочастотная метка находится в состоянии **ready** или **killed**, в этом случае она должна игнорировать команду *ACK* и оставаться в своем предшествующем состоянии.

Таблица 55 — Команда *ACK*

| Свойство | Команда | RN |
|---|---|---|
| Число битов | 2 | 16 |
| Описание | 01 | Повтор кода StoredCRC или маркера подлинности |

Таблица 56 — Ответ радиочастотной метки на «успешную» команду *ACK*

| Свойство | Ответ |
|---|---|
| Число битов | 5 до 528 |
| Описание | См. таблицу 42 |

Метод PJM: См. 6.3.3.3.4.8 и 6.3.3.3.4.11.

УСО могут одновременно принимать ответы вплоть до восьми радиочастотных меток одновременно. Этот режим позволяет для всего множества отвечающих радиочастотных меток выполнить общую адресацию в одной команде с расширенным полем *RN,* включающим в себя множество кодов Stored*CRC* или маркеров подлинности. Для подтверждения радиочастотной метки или меток этот режим использует команду *ACK-PJM,* приведенную на рисунке 40. Эта команда содержит одно или более значений кодов StoredCRC или маркеров подлинности. Выбор кода StoredCRC или маркера подлинности должен выполняться, как описано выше для Метода ASK.

Одна или более радиочастотных меток должны отвечать на «успешную» команду *ACK-PJM,* как показано в таблице 56. Ответ радиочастотной метки или меток на команду *ACK-PJM* должен быть таким, как описано выше для Метода ASK.

Таблица 57 — Команда *ACK-PJM*

| Свойство | Команда | RN | CRC-5 |
|---|---|---|---|
| Число битов | 2 | Количество радиочастотных меток для подтверждения умножить на 16 | 5 |
| Описание | 01 | Повтор кодов StoredCRC или маркеров подлинности | — |

6.3.3.4.11.2.5 Команда *NAK* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *NAK*, приведенной в таблице 58. Команда *NAK* должна возвращать все радиочастотные метки в состояние **arbitrate,** за исключением меток, находящихся в состоянии **ready** или **killed,** в этом случае они должны игнорировать команду *NAK* и сохранять свое текущее состояние.

Команде *NAK* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

Радиочастотные метки не должны отвечать на команду *NAK*.

Таблица 58 — Команда *NAK*

| Свойство | Команда |
|---|---|
| Число битов | 8 |
| Описание | 11000000 |

6.3.3.4.11.3 Команды доступа

Набор команд доступа включает в себя команды *Req_RN, Read, Write, Kill, Lock, Access, BlockWrite, BlockErase и BlockPermalock*. Как описано в 6.3.3.4.9, радиочастотная метка выполняет команду *Req_RN* в состояниях **acknowledged, open** или **secured**. Команды *Read, Write, BlockWrite* и *BlockErase* радиочастотной метки выполняются в состоянии **secured**; если статусом блокировки разрешено обращение к области памяти, то можно также выполнять эти команды в состоянии **open**. Команды *Access* и *Kill* радиочастотные метки выполняют в состояниях **open** или **secured**. Команды *Lock* и *BlockPermalock* радиочастотные метки выполняют только в состоянии **secured**.

Все команды доступа, поданные на радиочастотную метку в состояниях **open** или **secured,** содержат маркер подлинности радиочастотной метки как параметр в команде. Находясь в этих состояниях, радиочастотные метки перед выполнением команды доступа должны убедиться, что маркер подлинности правильный, и игнорировать команды доступа при неправильном маркере подлинности. Значение маркера подлинности неизменно на все время выполнения последовательности команд доступа.

Радиочастотная метка отвечает на все команды доступа, которые читают или записывают в память (т. е., *Read, Write, Kill, Lock, BlockWrite, BlockErase* и *BlockPermalock*), включая в ответ 1 бит заголовка. Заголовок=0 показывает, что операция была выполнена успешно и ответ действителен; заголовок=1 показывает, что операция была выполнена неуспешно и ответ — это код ошибки.

После того как УСО запишет в радиочастотную метку код StoredCRC или UII, либо изменит биты с $03_h$ по $07_h$ Пользовательской памяти с нуля на единичное значение (или наоборот), либо «введет в действие» радиочастотную метку, коды StoredCRC могут оказаться неверными до тех пор, пока УСО не выключит — включит питание радиочастотного поля, потому что радиочастотная метка рассчитывает свои коды StoredCRC по включении питания.

Если УСО делает попытку записи в память UII по адресам с $\mathbf{00_h}$ по $\mathbf{0F_h}$ (т. е., в область кода StoredCRC), используя команду *Write, BlockWrite* или *BlockErase*, радиочастотная метка должна игнорировать эту команду и ответить кодом ошибки (см. приложение Е для определения кодов ошибки и формата ответа).

Если УСО делает попытку записи в память UII по адресам с $\mathbf{210_h}$ по $\mathbf{22F_h}$ (т. е., в область XPC_W1 или XPC_W2), используя команду *Write, BlockWrite* или *BlockErase*, радиочастотная метка должна игнорировать эту команду и ответить кодом ошибки (см. приложение Е для определения кодов ошибки и формата ответа).

Если УСО делает попытку записи в банк памяти или пароль, которые «постоянно блокированы» для записи, или «заблокированы» для записи, при этом радиочастотная метка не находится в состоянии **secured**, или в блок памяти, находящийся в состоянии «постоянной блокировки», используя команду *Write, BlockWrite* или *BlockErase*; либо если сегмент Данные в команде *Write* указывает на блок памяти с «постоянной блокировкой», радиочастотная метка должна игнорировать эту команду и ответить кодом ошибки (см. приложение Е для определения кодов ошибки и формата ответа).

Для выполнения команд *Req_RN, Read, Write, Kill* (только с активированным третьим младшим значащим битом параметра Recom) требуются команды *Lock*; *Access*, *BlockWrite*, *BlockErase* и опционально *BlockPermalock*. Радиочастотные метки должны игнорировать опциональные команды доступа, которые они не поддерживают.

В приложении К показан пример обмена данными для случая, когда УСО выполняет доступ к радиочастотной метке и читает ее пароль уничтожения.

6.3.3.4.11.3.1 Команда *Req_RN* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *Req_RN*, приведенной в таблице 59. Команда *Req_RN* указывает радиочастотной метке передать в ответе новое значение RN16. Вид команды УСО и ответа радиочастотной метки зависят от текущего состояния радиочастотной метки:

- Состояние **acknowledged**: Когда передается команда *Req_RN* для радиочастотной метки в состоянии **acknowledged**, УСО должно включить код StoredCRC радиочастотной метки, как параметр,

в команду *Req_RN*. Команда *Req_RN* защищена кодом CRC-16c, рассчитанным относительно битов команды и кода StoredCRC. Если радиочастотная метка получает команду *Req_RN* с правильным значением CRC-16c и правильным кодом StoredCRC, то она должна сгенерировать и сохранить значение RN16 (означающее маркер подлинности), передать ответ с этим маркером подлинности и перейти в состояние **open** или **secured**. Выбор конечного состояния зависит от значения *пароля доступа* радиочастотной метки следующим образом:

- Пароль доступа <> 0: Радиочастотная метка переходит в состояние **open**.
- Пароль доступа = 0: Радиочастотная метка переходит в состояние **secured**.

Если радиочастотная метка получает команду *Req_RN* с правильным значением CRC-16c, но с неверным *кодом StoredCRC*, она должна игнорировать эту команду и остаться в состоянии **secured**.

- Состояния **open** или **secured**: Когда передается команда *Req_RN* для радиочастотной метки в состояниях **open** или **secured**, УСО должно включить маркер подлинности радиочастотной метки как параметр в команду *Req_RN*. Команда *Req_RN* защищена кодом CRC-16c, рассчитанным относительно битов команды и маркера подлинности. Если радиочастотная метка получает команду Req_RN с правильным кодом CRC-16c и правильным маркером подлинности, она должна сгенерировать новый RN16 и передать его в ответе. Если радиочастотная метка получает команду *Req_RN* с правильным кодом CRC-16c и неверным маркером подлинности, она должна игнорировать эту команду. В любом случае радиочастотная метка должна оставаться в своем текущем состоянии (**open** или **secured**).

Если УСО хочет гарантировать то, что только одна радиочастотная метка находится в состоянии **acknowledged**, она может дать команду *Req_RN*, заставляющую радиочастотную метку или каждую из меток передать в ответ маркер подлинности и перейти в состояние **open** или **secured** (по обстоятельствам). Затем УСО может дать команду *ACK* с маркером подлинности как параметром. Радиочастотные метки, принявшие команду *ACK* с неверным маркером подлинности, должны вернуться в состояние **arbitrate.**

Примечание — Если радиочастотная метка принимает команду *ACK* с неверным маркером подлинности, она возвращается в состояние arbitrate, тогда как если она принимает команду доступа с неверным маркером подлинности, то она игнорирует эту команду.

Первый бит передаваемого значения RN16 должен быть его СЗР (старшим значащим битом); последний бит должен быть его МЗР (младшим значащим битом).

Команде *Req_RN* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

Ответ радиочастотной метки на команду *Req_RN* должен быть таким, как показано в таблице 60. *Код StoredCRC* и маркер подлинности защищены CRC-16c.

Таблица 59 — Команда *Req_RN*

| Свойство | Команда | RN | CRC-16c |
|---|---|---|---|
| Число битов | 8 | 16 | 16 |
| Описание | 11000001 | Код StoredCRC или маркер подлинности | — |

Таблица 60 — Ответ радиочастотной метки на команду *Req_RN*

| Свойство | RN | CRC-16c |
|---|---|---|
| Число битов | 16 | 16 |
| Описание | Новое значение RN16 или маркер подлинности | — |

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. По вопросам, касающимся команды *ACK-PJM*, см. 6.3.3.4.11.2.4 и таблицу 57. Команда *ACK-PJM* может содержать множество *кодов StoredCRC* или маркеров подлинности в поле *RN*, чтобы адресоваться к множеству радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании данной команды) так же, как было описано выше для метода ASK, за исключением того, что данная команда может быть адресована множеству радиочастотных меток, и отвечать на данную команду может также множество радиочастотных меток.

6.3.3.4.11.3.2 Команда *Read* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *Read*, приведенной в таблице 62. *Read* позволяет УСО считывать частично или полностью резервную, UII, TID или Пользовательскую память радиочастотной метки. *Read* имеет следующие поля:

- MemBank определяет, к какой памяти команда *Read* будет иметь доступ: к резервной, UII, TID или Пользовательской памяти. Команда *Read* должна применяться только к одному банку памяти. Последующие команды *Read* могут применяться к другим банкам.

- WordPtrlength, WordPtr определяют адрес начального слова области для считывания памяти, где слова имеют длину 16 бит. Например, WordPtr = $00_h$ указывает первое 16-битовое слово в памяти, WordPtr = $01_h$ указывает второе 16-битовое слово в памяти и т. д. Параметр WordPtr имеет длину 8, 16, 24 или 32 бита, как определено в WordPtrlength.

- WordCount указывает число 16-битовых слов для считывания. Если поле WordCount = $00_h$, радиочастотная метка должна передавать содержимое выбранного банка памяти, начиная с WordPtr и заканчивая концом банка, за исключением случая, когда MemBank = $01_2$, в этом случае радиочастотная метка должна передавать содержимое памяти UII, указанное в таблице 61.

Таблица 61 — Передача радиочастотной метки для случая, когда WordCount=$00_h$ и MemBank=$01_2$

| Адрес памяти WordPtr | Метка поддерживает XPC_W2? | Что передает радиочастотная метка |
|---|---|---|
| В пределах кодов StoredCRC, данных StoredPC или UII, указанных в битах от $10_h$ до $14_h$ данных StoredPC | Не контролирует | Память UII, начиная с WordPtr и заканчивая по значению длина UII, указанному в битах $10_h$ – $4_h$ данных StoredPC |
| В пределах физической памяти UII, но выше значения длина UII, указанного в битах от $10_h$ до $14_h$ данных StoredPC | Да | Память UII, начиная с WordPtr и заканчивая концом физической памяти UII, включая XPC_W1 и XPC_W2, если WordPtr меньше или равен $210_h$, расширенной физической памяти UII для или выше адреса $210_h$ включая XPC_W2, если WordPtr равен $220_h$ и расширенной физической памяти UII для или выше адреса $220_h$ |
| $210_h$. Верхняя физическая память UII | Нет | XPC_W1 |
| $210_h$. Верхняя физическая память UII | Да | XPC_W1 и XPC_W2 |
| $220_h$. Верхняя физическая память UII | Нет | Код ошибки |
| $220_h$. Верхняя физическая память UII | Да | XPC_W2 |
| Не $210_h$ или $220_h$. Верхняя физическая память UII | Не контролирует | Код ошибки |

Команда *Read* также содержит маркер подлинности радиочастотной метки и CRC-16c. CRC-16 рассчитывается от первого бита кода команды до последнего бита маркера подлинности.

Если радиочастотная метка принимает команду *Read* с правильным CRC-16c, но неверным маркером подлинности, она должна игнорировать эту команду и сохранять свое текущее состояние (**open** или **secured** по обстоятельствам).

Команде *Read* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

Если все слова памяти, указанные в команде *Read*, существуют и не блокированы для считывания, ответ радиочастотной метки на команду *Read* должен быть такой, как приведено в таблице 63. Радиочастотная метка отвечает, передавая заголовок (бит 0), запрошенные слова памяти и свой маркер подлинности. Ответ включает в себя CRC-16c, рассчитанный от 0 бита области памяти до маркера подлинности.

Если одно или более слов памяти, указанных в команде *Read*, либо не существуют, либо заблокированы для считывания, радиочастотная метка должна передать код ошибки внутри интервала $T_1$, как показано в таблице 41, предпочтительнее, чем ответ, приведенный в таблице 63 (см. приложение Е для определения кодов ошибки и формата ответа).

Таблица 62 — Команда *Read*

| Свойство | Команда | MemBank | WordPtr-Length | WordPtr | WordCount | RN | CRC-16c |
|---|---|---|---|---|---|---|---|
| Число битов | 8 | 2 | 2 | 8, 16, 24 или 32 | 8 | 16 | 16 |
| Описание | 11000010 | 00: Резервный<br>01: UII<br>10: TID<br>11: Пользовательский | Длина указателя слова<br>00: 8 бит<br>01: 16 бит<br>10: 24 бит<br>11: 32 бит | Указатель начального адреса | Число слов для считывания | Маркер подлинности | — |

Таблица 63 — Ответ радиочастотной метки на успешную команду *Read*

| Свойство | Заголовок | Слова памяти | RN | CRC-16c |
|---|---|---|---|---|
| Число битов | 1 | Переменная | 16 | 16 |
| Описание | 0 | Данные | Маркер подлинности | — |

Опционально УСО должны устанавливать связь, используя метод PJM.

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Также для метода *ACK-PJM* см. 6.3.3.4.11.2.4 и таблицу 57. Данная команда может содержать множество маркеров подлинности в поле *RN* для адресации множества радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании данной команды), как описано выше для метода ASK, за исключением того, что данная команда может адресоваться множеству радиочастотных меток, и множество радиочастотных меток может отвечать на эту команду.

6.3.3.4.11.3.3 Команда *Write* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *Write*, приведенной в таблице 64. Команда *Write* позволяет УСО записывать слова в Резервную, UII, TID или Пользовательскую память радиочастотной метки. Команда *Write* имеет следующие поля:

- MemBank, которое определяет, будет ли команда Write иметь доступ к Резервной, UII, TID или Пользовательской памяти.

- WordPtrlength, WordPtr, которые определяют адрес начального слова области для записи в память, где слова имеют дину 16 бит. Например, WordPtr = $00_h$ указывает первое 16-битовое слово в памяти, WordPtr = $01_h$ указывает второе 16-битовое слово в памяти, и т. д. Параметр WordPtr имеет длину 8, 16, 24 или 32 бита, как определено в WordPtrlength.

- Данные, включающие 16-битовое слово для записи. Дополнительно, перед каждой командой *Write* УСО может подать команду *Req_RN*; радиочастотная метка ответит передачей нового значения RN16. Если была дана команда *Req_RN*, то УСО должно использовать защиту кодированием данных путем выполнения перед передачей операции ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR) с новым значением RN16.

Команда *Write* также содержит маркер подлинности радиочастотной метки и CRC-16c. CRC-16 рассчитывается от первого бита кода команды до последнего бита маркера подлинности. Если радиочастотная метка находится в состоянии **open** или **secured** и принимает команду *Write* с правильным CRC-16c, но с неверным маркером подлинности, она должна игнорировать команду *Write* и оставаться в текущем состоянии. Если она принимает команду *Write*, причем непосредственно предшествующей командой не была команда *Req_RN*, то она должна интерпретировать поле данных как непосредственные данные.

Команде *Write* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8). После подачи команды *Write* УСО должно передать CW длительностью наименьшей из значений: $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *Write* и передачей ответа радиочастотной меткой. Время $T_{ОТВЕТА}$ необходимо умножить на типовое значение $T_1$ (см. таблица 41) с отклонением ± 2,4 мкс, так что $T_{ОТВЕТА}$ = n*1024/fc ± 32/fc. УСО может иметь несколько возможных исходов выполнения команды *Write*, зависящих от успешности или сбоя выполнения записи в память радиочастотной метки:

- **Команда Write выполнена успешно:** После завершения команды Write радиочастотная метка должна передать ответ, приведенный в таблице 65, и на рисунке 46, включающий заголовок (бит 0),

маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что команда Write завершена успешно;

**- Радиочастотная метка обнаружила ошибку:** Радиочастотная метка должна передать ответ с кодом ошибки в течение интервала CW вместо ответа, приведенного на рисунке 46 (см. приложение Е для определения кодов ошибок и формата ответа);

**- Команда Write не выполнена:** Если УСО получает этот ответ в интервале 20 мс, то команда *Write* не была завершена успешно. УСО может подать команду *Req_RN* (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, если использовалась защита кодированием, или может повторно подать команду *Write*, независимо от использования защиты кодированием.

После приема действительной команды *Write* радиочастотная метка должна записать указанные командой данные в память. Ответ радиочастотной метки на успешную команду *Write* должен использовать такой же заголовок, как определенный в команде *BeginRound*, которая инициирует цикл.

Таблица 64 — Команда *Write*

| Свойство | Команда | MemBank | WordPtr-Length | WordPtr | Данные | RN | CRC-16c |
|---|---|---|---|---|---|---|---|
| Число битов | 8 | 2 | 2 | 8, 16, 24 или 32 | 16 | 16 | 16 |
| Описание | 11000011 | 00: Резервный<br>01: UII<br>10: TID<br>11: Пользователь | Длина указателя слова<br>00: 8 битов<br>01: 16 битов<br>10: 24 бита<br>11: 32 бита | Адрес указателя | Будет записано RN16 ⊗ слово данных или слово данных | Маркер подлинности | — |

Таблица 65 — Ответ радиочастотной метки на успешную команду *Write*

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | — |




Рисунок 46 — Временная диаграмма успешной команды *Write*

Дополнительно УСО должны устанавливать связь, используя метод PJM.

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Для *ACK-PJM* см. 6.3.3.4.11.2.4 и таблицу 57. Данная команда может содержать множество маркеров подлинности в поле RN для адресации множества радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании этой команды), как описано выше для метода ASK, за исключением того, что данная команда может быть адресована множеству радиочастотных меток, и множество радиочастотных меток могут отвечать на эту команду.

6.3.3.4.11.3.4 Команда *Kill* (обязательна только поддержка активированного третьего МЗР поля Recom)

Метод ASK и Метод PJM: УСО и радиочастотные метки должны включать в себя средства реализации команды *Kill* (обязательна только поддержка активированного третьего МЗР поля Recom), приведенной в таблице 66 и на рисунке 47. *Kill* позволяет УСО навсегда отключить («уничтожить») радиочастотную метку. *Kill* также позволяет УСО «ввести в действие» радиочастотную метку.

Для «уничтожения» или «ввода в строй» радиочастотной метки УСО должно следовать многошаговой процедуре, изложенной на рисунке 47. УСО выдает две команды *Kill*, первая содержит 16 старших

значащих битов «пароля уничтожения» радиочастотной метки, при необходимости замаскированные операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с RN16, а вторая содержит 16 младших значащих битов «пароля уничтожения» радиочастотной метки, при необходимости замаскированные операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с другим RN16. Каждая операция «ИСКЛЮЧАЮЩЕЕ ИЛИ» должна быть выполнена старшим значащим битом вперед (т. е. СЗР каждой половины пароля должен маскироваться операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с СЗР соответствующего ему RN16). Если используется защита кодированием, перед подачей каждой команды *Kill* УСО сначала подает команду *Req_RN* для получения нового RN16, в противном случае команда *Req_RN* не подается.

Радиочастотные метки должны включать в себя необходимую логику для успешного приема двух 16-битовых частей 32-битового «пароля уничтожения». УСО не должны подавать другие команды, кроме *Req_RN* между двумя следующими друг за другом командами *Kill*. Если радиочастотная метка после приема первой команды *Kill* примет любую другую команду, кроме *Req_RN* перед второй командой *Kill*, то она должна вернуться в состояние **arbitrate;** за исключением поступления команды *BeginRound*, в этом случае радиочастотная метка должна выполнить команду *BeginRound* (установить свой флаг **inventoried**, если параметр сеанса в команде *BeginRound* соответствует предыдущему сеансу).

Команда *Kill* содержит 3 бита RFU/Recom. В первой команде *Kill* эти биты RFU. УСО при соединении с радиочастотными метками должны установить в 3 бита RFU значение $000_2$, а радиочастотные метки должны игнорировать эти биты. Как описано в 6.3.3.4.10, во второй команде *Kill* 3 бита RFU называются битами «ввод в действие» (или Recom) и могут быть ненулевыми. Процедуры «уничтожения» или «ввода в действие» радиочастотной метки идентичны, за исключением того, что во второй команде *Kill* биты «ввода в действие» нулевые при «уничтожении» (необязательно) радиочастотной метки и ненулевые при «вводе в действие» (обязательна поддержка 3-го МЗР Recom) радиочастотной метки. Независимо от предполагаемой операции радиочастотная метка не должна отключаться или повторно «вводиться в действие» самостоятельно без предварительно принятого правильного «пароля уничтожения» по процедуре, показанной в 6.3.3.4.10.

Если радиочастотная метка не поддерживает первый и второй МЗР Recom (см. 6.3.3.4.10), то радиочастотная метка игнорирует первый и второй МЗР «ввода в действие» и обрабатывает их как будто они нулевые. Если радиочастотная метка принимает правильно форматированную командную последовательность *Kill* с правильным «паролем уничтожения», и все 3 МЗР Recom нулевые, а радиочастотная метка не поддерживает вторую команду *Kill* со всеми 3 битами Recom, установленными в ноль, поскольку это необязательно, то радиочастотная метка должна интерпретировать эти команды *Kill* так, как будто их три младших значащих бита Recom установлены в единицы. Если радиочастотная метка не поддерживает «уничтожение», то радиочастотная метка выполняет повторный «ввод в действие» самостоятельно, независимо от значений битов «ввода в действие», если «ввод в действие» разрешен.

Радиочастотная метка с нулевым «паролем уничтожения» не выполняет операции «уничтожения» или «ввода в действие». Если такая радиочастотная метка принимает команду *Kill*, она игнорирует эту команду и передает код ошибки (см. рисунок 47).

Ответ радиочастотной метки на первую команду *Kill* должен быть таким, как описано в таблице 68. Ответ должен использовать значение TRext, указанной в команде *BeginRound*, которая инициирует цикл.

После выдачи команды *Kill* УСО должно передать CW длительностью, наименьшей из значений $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *Kill* и передачей ответа радиочастотной меткой. Время $T_{ОТВЕТА}$ необходимо умножить на типовое значение $T_1$ (см. таблицу 41) с отклонением ±2,4 мкс, так что $T_{ОТВЕТА} = n*1024/f_c \pm 32/f_c$. УСО может иметь несколько возможных исходов выполнения команды *Kill*, зависящих от успешности или сбоя выполнения записи в память радиочастотной метки:

- «Уничтожение» или «ввод в действие» выполнены: После завершения операции радиочастотная метка должна передать ответ, приведенный в таблице 69 и на рисунке 47, включающий заголовок (бит 0), маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что операция завершена успешно. Если радиочастотная метка «уничтожена», то сразу же после этого ответа радиочастотная метка должна перевести себя в «скрытое» состояние и после этого не должна отвечать УСО.

- Радиочастотная метка обнаружила ошибку: Радиочастотная метка должна в течение интервала CW вместо ответа передать код ошибки, приведенный в таблице 69 (см. приложение Е для определения кодов ошибки и формата ответа).

- «Уничтожение» или «ввод в действие» не были выполнены успешно: Если УСО получит этот ответ в интервале 20 мс, то считает, что операция не была завершена успешно. Если использовалась защита кодированием, УСО может подать команду Req_RN (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, или может повторно инициироваться многошаговой процедурой *Kill*, показанной на рисунке 47, независимо от использования защиты кодированием. Команде *Kill* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

После приема действительной командной последовательности *Kill* радиочастотная метка должна перевести себя в «скрытое» состояние или состояние «введена в действие», по обстоятельствам. В ответе радиочастотной метки на вторую команду *Kill* должен использоваться заголовок, определенный в команде *BeginRound*, которая инициирует цикл.

Таблица 66 — Первая команда *Kill*

| Свойство | Команда | Пароль | RFU/ Recom | RN | CRC-16c |
|---|---|---|---|---|---|
| Число битов | 8 | 16 | 3 | 16 | 16 |
| Описание | 11000100 | ($^1/_2$ пароля уничтожения) ⊗ RN16 или пустой | 000 | Маркер подлинности | — |

Таблица 67 — Вторая команда *Kill*

| Свойство | Команда | Пароль | RFU/Recom | RN | CRC-16c |
|---|---|---|---|---|---|
| Число битов | 8 | 16 | 3 | 16 | 16 |
| Описание | 11000100 | ($^1/_2$ пароля уничтожения) ⊗ RN16 или пустой | Бит «ввода в действие» (см. 6.3.3.4.10) | Маркер подлинности | — |

Таблица 68 — Ответ радиочастотной метки на первую команду *Kill*

| Свойство | RN | CRC-16c |
|---|---|---|
| Число битов | 16 | 16 |
| Описание | Маркер подлинности | — |

Таблица 69 — Ответ радиочастотной метки на успешную процедуру *Kill*

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | |

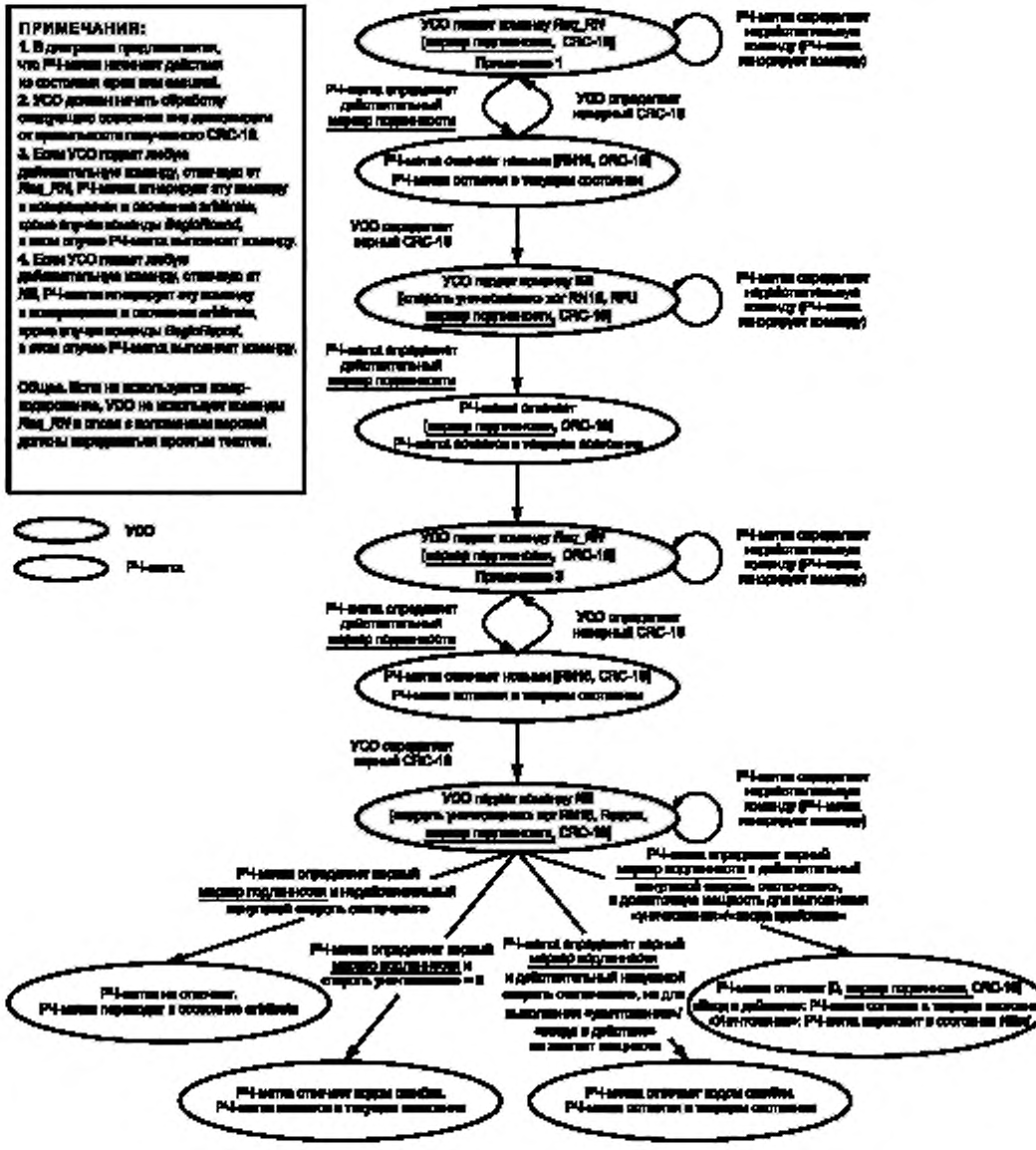



Рисунок 47 — Процедура *Kill*

6.3.3.4.11.3.5 Команда *Lock* (обязательная)

УСО и радиочастотные метки должны включать в себя средства реализации команды *Lock*, приведенной в таблице 70, и на рисунке 48. Команду *Lock* должны выполнять только радиочастотные метки, находящиеся в состоянии **secured**. Команда *Lock* позволяет УСО:

- блокировать определенные пароли, таким образом предотвращая или разрешая последующие считывание и запись этого пароля;
- блокировать определенные банки памяти, таким образом предотвращая или разрешая последующую запись в этот банк;
- выполнять «постоянную блокировку» (делать «постоянно неизменяемым») — состояние блокировки для пароля или банка памяти.

Команда *Lock* содержит 20-битовую строку параметров, определяемую следующим образом:

- Первые 10 бит строки параметров — это биты Mask. Радиочастотная метка должна интерпретировать значения этих битов следующим образом:
  - Mask = 0: Игнорировать соответствующее поле Action и сохранять текущую настройку блокировки;
  - Mask = 1: Реализовать соответствующее поле Action и перезаписать текущую настройку блокировки.
- Последние 10 бит строки параметров — это биты Action. Радиочастотная метка должна интерпретировать значения этих битов следующим образом:
  - Action = 0: Деактивировать блокировку для соответствующей области памяти;
  - Action = 1: Активировать блокировку или «постоянную блокировку» для соответствующей области памяти.

Назначение различных полей Action описано в таблице 72.

Строка параметров команды *Lock* должна всегда быть длиной 20 битов. Если УСО выдает команду *Lock*, в которой поля Mask и Action указывают на изменение состояния блокировки несуществующего банка памяти или несуществующего пароля, радиочастотная метка должна игнорировать всю команду *Lock* и вместо этого передавать код ошибки (см. приложение Е).

Команда *Lock* отличается от необязательной команды *BlockPermalock* тем, что команда *Lock* выполняет обратимую или «постоянную блокировку» пароля, банка памяти UII, TID или Пользовательской памяти в записываемое или незаписываемое состояние, тогда как команда *BlockPermalock* выполняет «постоянную блокировку» блоков Пользовательской памяти в незаписываемое состояние. Таблица 79 определяет, как радиочастотная метка реагирует на команду *Lock*, следующую за предшествующей командой *BlockPermalock*, или наоборот.

Однажды реализованные биты «постоянной блокировки» уже не могут быть отменены, за исключением случая повторного «ввода в действие» радиочастотной метки (см. 6.3.3.4.10). Если радиочастотная метка принимает команду *Lock*, строка параметров которой содержит сброшенный бит блокировки для бита, ранее реализованного для «постоянной блокировки», радиочастотная метка должна игнорировать такую команду *Lock* и выполнить передачу кода ошибки (см. приложение Е). Если радиочастотная метка принимает команду *Lock*, которая пытается повторно активировать ранее реализованный бит «постоянной блокировки», радиочастотная метка должна просто игнорировать ее поле Action и оставлять адресуемую память в состоянии блокировки, за исключением случая, когда радиочастотная метка была повторно «введена в действие», и адресуемая область перестала быть заблокированной. В этом случае в радиочастотной метке должна быть реализована «постоянная блокировка» по активированному биту.

Биты блокировки радиочастотной метки не могут быть прочитаны напрямую; они могут быть получены при попытке выполнить другие операции с памятью.

Все радиочастотные метки должны включать в себя средства реализации блокировки памяти, а также все радиочастотные метки должны выполнять команду *Lock*. Однако радиочастотным меткам нет необходимости поддерживать все поля Action, показанные на рисунке 48, в зависимости от того, существует ли место расположения пароля или банка памяти соответствующее полю Action и является ли оно блокируемым и/или неблокируемым. В частности, если радиочастотная метка принимает команду *Lock*, которую не может выполнить из-за того, что один или более паролей или банков памяти не существует, либо одно или более полей Action пытаются изменить ранее «постоянно заблокированное» состояние, либо один или более паролей или банков памяти являются неблокируемыми, либо неразблокируемыми. Радиочастотная метка должна игнорировать всю команду *Lock* и передавать код ошибки (см. приложение Е). Единственное исключение из этого общего правила относится к радиочастотной метке, которая была заблокирована «постоянной блокировкой» **всей** памяти (т. е., все банки памяти и все пароли) сразу; такие радиочастотные метки должны выполнять команду *Lock*, чья строка параметров имеет значение $FFFFF_h$, и должны передавать код ошибки для любой другой строки параметров кроме $FFFFF_h$.

Команде *Lock* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

После подачи команды *Lock* УСО должно передать CW длительностью наименьшей из значений: $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *Lock* и передачей ответа радиочастотной меткой. Время $T_{ОТВЕТА}$ необходимо умножить на типовое $T_1$ (см. таблицу 41) с отклонением ± 2,4 мкс, так что $T_{ОТВЕТА} = n*1024/fc \pm 32/fc$. УСО может иметь несколько возможных исходов

выполнения команды *Lock*, зависящих от успешности или сбоя выполнения записи в память радиочастотной метки.

- **Команда Lock выполнена успешно:** После завершения команды Lock радиочастотная метка должна передать ответ, приведенный в таблице 71 и на рисунке 46, включающий заголовок (бит 0), маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что команда Lock завершена успешно.

- **Радиочастотная метка обнаружила ошибку:** Радиочастотная метка должна передать код ошибки в течение интервала CW вместо ответа, приведенного в таблице 71 (см. приложение Е для определения кодов ошибки и формата ответа).

- **Команда Lock не была успешной:** Если УСО получает этот ответ в интервале 20 мс, то считает, что команда Lock не была завершена успешно. УСО может подать команду Req RN (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, или может повторно выдать команду Lock.

После приема действительной команды *Lock* радиочастотная метка должна выполнить предписанную операцию блокировки. Ответ радиочастотной метки на команду *Lock* должен использовать заголовок, как указано в команде *BeginRound*, инициировавший цикл.

Таблица 70 — Команда *Lock*

| Свойство | Команда | Строка параметров | RN | CRC-16c |
|---|---|---|---|---|
| Число битов | 8 | 20 | 16 | 16 |
| Описание | 11000101 | Поля Mask и Action | Маркер подлинности | — |

Таблица 71 — Ответ радиочастотной метки на команду *Lock*

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | — |

**Строка праметров команды Lock**

| 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| «Уничтожение» Mask | | Доступа Mask | | UII Mask | | TID Mask | | Пользовательская Mask | | «Уничтожения» Action | | Доступа Action | | UII Action | | TID Action | | Пользовательская Action | |

**Поля Mask и связанные с ними поля Action**

| | Пароль «уничтожения» | | Пароль доступа | | Память UII | | Память TID | | Пользовательская память | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 |
| Mask | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи | Пропуск записи |
| | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Action | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи | Пароль чтения/ записи |

Рисунок 48 — Содержание строки параметров команды *Lock* и их применение

Таблица 72 — Значение полей Action команды *Lock*

| «Пароль записи» | «Постоянная блокировка» | Описание |
|---|---|---|
| 0 | 0 | Соответствующий банк памяти, записываемый в состоянии **open** или **secured** |
| 0 | 1 | Соответствующий банк памяти, постоянно записываемый в состояниях **open** или **secured**, никогда не может быть заблокированным |
| 1 | 0 | Соответствующий банк памяти, записываемый в состоянии **secured,** но не в состоянии **open** |
| 1 | 1 | Соответствующий банк памяти, незаписываемый ни в одном из состояний |
| 0 | 0 | Соответствующая ячейка пароля, считываемая и записываемая в состоянии **open** или **secured** |
| 0 | 1 | Соответствующая ячейка пароля, постоянно считываемая и записываемая в состоянии **open** или **secured,** никогда не может быть заблокирована |
| 1 | 0 | Соответствующая ячейка пароля, считываемая и записываемая в состоянии **secured**, но не в состоянии **open** |
| 1 | 1 | Соответствующая ячейка пароля, несчитываемая или незаписываемая в любом состоянии |

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Как для *ACK-PJM* (см. 6.3.3.4.11.2.4 и таблицу 57), данная команда может содержать множество маркеров подлинности в поле RN для адресации множества радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании данной команды), как описано выше для метода ASK, за исключением того, что данная команда может быть адресована множеству радиочастотных меток, и множество радиочастотных меток могут отвечать на данную команду.

6.3.3.4.11.3.6 Команда *Access* (необязательная)

Метод ASK и Метод PJM: УСО и радиочастотные метки могут включать в себя средства реализации команды *Access*; если это так, то команда должна иметь вид, приведенный в таблице 73. *Access* заставляет радиочастотную метку с ненулевым значением пароля доступа перейти из состояния **open** в состояние **secured** (радиочастотная метка с нулевым значением пароля доступа никогда не находится в состоянии **open** — см. рисунок 43), либо для сохранения состояния **secured**, если радиочастотная метка уже находится в состоянии **secured.**

Для реализации доступа к радиочастотной метке (команды *Access)* УСО должно следовать многошаговой процедуре, отображенной на рисунке 49. Кратко, УСО выдает две команды *Access*, первая содержит 16 старших значащих битов «пароля доступа» радиочастотной метки при желании замаскированных операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с RN16, а вторая содержит 16 младших значащих битов «пароля доступа» радиочастотной метки при желании замаскированное операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с другим RN16. Каждая операция «ИСКЛЮЧАЮЩЕЕ ИЛИ» должна быть выполнена СЗР вперед (т. е. СЗР каждой половины пароля должен маскироваться операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» с СЗР соответствующего ей RN16). Если используется защита кодированием, перед подачей каждой команды *Access* УСО сначала подает команду *Req_RN* для получения нового RN16, в противном случае команда *Req_RN* не используется.

Радиочастотные метки должны включать в себя необходимую логику для успешного приема двух 16-битовых частей 32-битового пароля доступа. Если используется защита кодированием, то УСО не должны выдавать другие команды кроме *Req_RN* между двумя следующими друг за другом командами *Access*. Если радиочастотная метка после приема первой команды *Access* примет любую другую команду, кроме *Req_RN* перед второй командой *Access*, то она должна вернуться в состояние **arbitrate,** за исключением случая поступления команды *BeginRound*, в этом случае радиочастотная метка должна выполнить команду *BeginRound* (установить свой флаг **inventoried**, если параметр сеанса в команде *BeginRound* соответствует предыдущему сеансу).

Команде *Access* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4 и 6.3.3.3.1.6 и 6.3.3.3.1.3.8).

Ответ радиочастотной метки на первую команду *Access* должен быть таким, как описано в таблице 74. Если команда *Access* первая в последовательности, то радиочастотная метка выполняет передачу ее маркера подлинности для того, чтобы подтвердить получение команды. Если команда *Access* вторая в последовательности и полный принятый 32-битовый «пароль доступа» верный, то радиочастотная метка передает в ответе свой маркер подлинности для того, чтобы подтвердить успешное выполнение команды и перейти в состояние **secured**; в ином случае радиочастотная метка не отвечает. Ответ содержит CRC-16c, рассчитанный по маркеру подлинности.

Таблица 73 — Команда *Access*

| Свойство | Команда | Пароль | RN | CRC-16c |
|---|---|---|---|---|
| Число битов | 8 | 16 | 16 | 16 |
| Описание | 11000110 | ($^1/_2$ пароля доступа) ⊗ RN16 или пустой | Маркер подлинности | — |

Таблица 74 — Ответ радиочастотной метки на команду *Access*

| Свойство | RN | CRC-16c |
|---|---|---|
| Число битов | 16 | 16 |
| Описание | Маркер подлинности | — |

6.3.3.4.11.3.7 Команда *BlockWrite* (необязательная)

УСО и радиочастотные метки могут включать в себя средства реализации команды *BlockWrite*; если это так, то команда должна иметь вид, приведенный в таблице 75. Команда *BlockWrite* позволяет УСО записать несколько слов в Резервную, UII, TID или Пользовательскую память радиочастотной метки, используя одну команду. Команда *BlockWrite* имеет следующие поля:

- MemBank, которое определяет, к какой памяти команда *BlockWrite* будет иметь доступ: к резервной, UII, TID или Пользовательской. Команда *BlockWrite* должна применяться только к одному банку памяти. Последующие команды *BlockWrite* могут применяться к другим банкам.

- WordPtrlength, WordPtr, которые определяют адрес начального слова области для записи в память, где слова имеют дину 16 битов. Например, WordPtr = $00_h$ указывает первое 16-битовое слово в памяти, WordPtr = $01_h$ указывает второе 16-битовое слово в памяти и т. д. Параметр WordPtr имеет длину 8, 16, 24 или 32 бита, как определено в WordPtrlength.

- WordCount, которое указывает число 16-битовых слов для записи. Если WordCount = $00_h$, радиочастотная метка должна игнорировать команду *BlockWrite*. Если WordCount = $01_h$, радиочастотная метка должна записать одно слово данных.

- Данные, которые хранятся в 16-битовых словах для записи и должны иметь длину, указанную в WordCount * 16 битов. В отличие от команды *Write*, данные в команде *BlockWrite* не подвергаются защите кодированием, и УСО нет необходимости подавать команду *Req_RN* перед подачей команды *BlockWrite*.

Команда *BlockWrite* также содержит маркер подлинности радиочастотной метки и CRC-16c. CRC-16 рассчитывается от первого бита кода команды до последнего бита маркера подлинности.

Если радиочастотная метка принимает команду *BlockWrite* с действительным CRC-16c, но неверным маркером подлинности, она должна игнорировать команду *BlockWrite* и сохранять свое текущее состояние **(open** или **secured,** по обстоятельствам). Команде *BlockWrite* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

После подачи команды *BlockWrite* УСО должно передать CW длительностью наименьшей из значений: $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *BlockWrite* и передачей ответа радиочастотной меткой. Время $T_{ОТВЕТА}$ необходимо умножить на типовое значение $T_1$ (см. таблицу 41) с отклонением ±2,4 мкс, так что $T_{ОТВЕТА}$ = n*1024/fc ± 32/fc. УСО может иметь несколько возможных исходов выполнения команды *BlockWrite*, зависящих от успешности или сбоя выполнения записи в память радиочастотной метки:

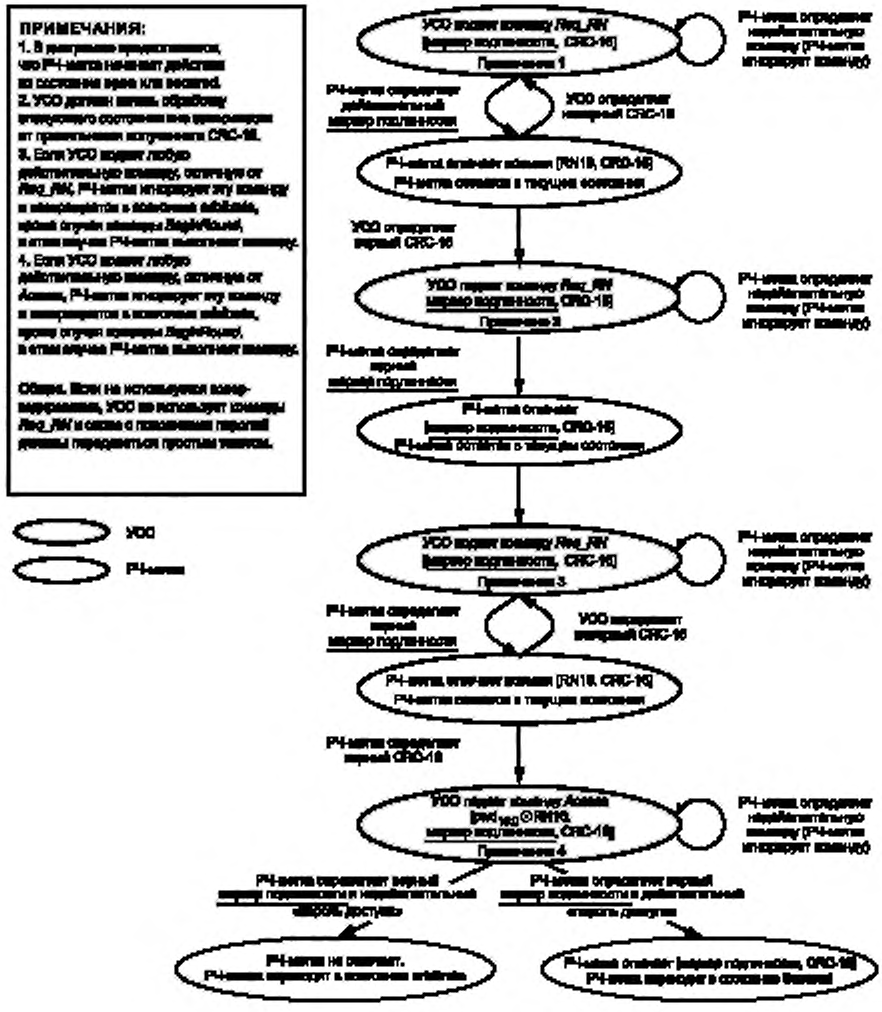



Рисунок 49 — Команда *Access*

- **Команда *BlockWrite* выполнена успешно:** После завершения команды *BlockWrite* радиочастотная метка должна передать ответ, приведенный в таблице 76, и на рисунке 46, включающий заголовок (бит 0), маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockWrite* завершена успешно.

- **Радиочастотная метка обнаружила ошибку:** Радиочастотная метка должна передать код ошибки в течение интервала CW вместо ответа, приведенного в таблице 76 (см. приложение Е для определения кодов ошибки и формата ответа).

- **Команда *BlockWrite* не была успешной:** Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockWrite* не была завершена успешно. УСО может подать команду *Req_RN* (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, или может повторно выдать команду *BlockWrite*.

После приема действительной команды *BlockWrite* радиочастотная метка должна записать указанные в ней данные в память. Ответ радиочастотной метки на успешную команду *BlockWrite* должен использовать заголовок, определенный в команде *BeginRound*, которая инициирует цикл.

Таблица 75 — Команда *BlockWrite*

| Свойство | Команда | MemBank | WordPtr-- Length | WordPtr | WordCount | Данные | RN | CRC-16c |
|---|---|---|---|---|---|---|---|---|
| Число битов | 8 | 2 | 2 | 8, 16, 24 или 32 | 8 | Переменная | 16 | 16 |
| Описание | 11000111 | 00: Резервная<br>01: UII<br>10: TID<br>11: Пользовательская | Длина указателя в словах<br>00: 8 битов<br>01: 16 битов<br>10: 24 бита<br>11: 32 бита | Указатель начального адреса | Количество слов для записи | Данные для записи | Маркер подлинности | — |

Таблица 76 — Ответ радиочастотной метки на успешную команду *BlockWrite*

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | — |

УСО могут устанавливать связь, используя метод PJM.

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Как для команды *ACK-PJM* (см. 6.3.3.4.11.2.4 и таблицу 57), данная команда может содержать несколько маркеров подлинности в поле RN для адресации нескольких радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании данной команды), как описано выше для метода ASK, за исключением того, что данная команда может быть адресована множеству радиочастотных меток, и множество радиочастотных меток могут отвечать на данную команду.

6.3.3.4.11.3.8 Команда *BlockErase* (необязательная)

УСО и радиочастотные метки могут включать в себя средства реализации команды *BlockErase*; если это так, то она должна иметь вид, приведенный в таблице 77. Команда *BlockErase* имеет следующие поля:

- MemBank определяет, к какой памяти команда *BlockErase* будет иметь доступ: к резервной, UII, TID или пользовательской. Команда *BlockErase* должна применяться только к одному банку памяти. Последующие команды *BlockErase* могут применяться к другим банкам.

- WordPtrlength, WordPtr определяют адрес начального слова области для очистки памяти, где слова имеют длину 16 битов. Например, WordPtr = $00_h$ указывает первое 16-битовое слово в памяти, WordPtr = $01_h$ указывает второе 16-битовое слово в памяти и т. д. Параметр WordPtr имеет длину 8, 16, 24 или 32 бита, как определено в WordPtrlength

- WordCount указывает число 16-битовых слов для записи. Если WordCount = $00_h$, радиочастотная метка должна игнорировать команду *BlockErase*. Если WordCount = $01_h$, радиочастотная метка должна удалить одно слово данных.

Команда *BlockErase* также содержит маркер подлинности радиочастотной метки и CRC-16c. CRC-16 рассчитывается от первого бита кода команды до последнего бита маркера подлинности.

Если радиочастотная метка принимает команду *BlockErase* с действительным CRC-16c, но неверным маркером подлинности, она должна игнорировать команду *BlockErase* и сохранять свое текущее состояние (**open** или **secured**, по обстоятельствам).

Команде *BlockErase* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.3.4, 6.3.3.3.1.3.6 и 6.3.3.3.1.3.8).

После подачи команды *BlockErase* УСО должно передать CW длительностью наименьшей из значений: $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *BlockErase* передачей ответа радиочастотной меткой. Время $T_{ОТВЕТА}$ необходимо умножить на типовое значение $T_1$ (см. таблицу 41) с отклонением ±2,4 мкс, так что $T_{ОТВЕТА} = n*1024/fc \pm 32/fc$. УСО может иметь несколько возможных исходов выполнения команды *BlockErase*, зависящих от успешности или сбоя выполнения очистки памяти радиочастотной метки:

- **Команда *BlockErase* выполнена успешно:** После завершения команды *BlockErase* радиочастотная метка должна передать ответ, приведенный в таблице 65 и на рисунке 46, включающий

заголовок (бит 0), маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockErase* завершена успешно.

- **Радиочастотная метка обнаружила ошибку:** Радиочастотная метка должна передать код ошибки в течение интервала CW вместо ответа, приведенного на рисунке 46 (см. приложение С для определения кодов ошибки и формата ответа).

- **Команда *BlockErase* не была успешной:** Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockErase* не была завершена успешно. УСО может подать команду *Req RN* (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, или может повторно выдать команду *BlockErase*.

После приема действительной команды *BlockErase* радиочастотная метка должна очистить указанные слова в памяти. Ответ радиочастотной метки на успешную команду *BlockErase* должен использовать заголовок, определенный в команде *BeginRound*, которая инициирует цикл.

Таблица 77 — Команда *BlockErase*

| Свойство | Команда | MemBank | WordPtr Length | WordPtr | WordCount | RN | CRC-16c |
|---|---|---|---|---|---|---|---|
| Число битов | 8 | 2 | 2 | 8, 16, 24 или 32 | 8 | 16 | 16 |
| Описание | 11001000 | 00: Резервная<br>01: UII<br>10: TID<br>11: Пользовательская | Длина указателя в словах<br>00: 8 битов<br>01: 16 битов<br>10: 24 бита<br>11: 32 бита | Указатель начального адреса | Количество слов для записи | Маркер подлинности | — |

Таблица 78 — Ответ радиочастотной метки на успешную команду *BlockErase*

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | — |

УСО могут устанавливать связь, используя метод PJM.

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Как для команды *ACK-PJM* (см. 6.3.3.4.11.2.4 и таблицу 57), данная команда может содержать несколько маркеров подлинности в поле RN для адресации нескольких радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании данной команды), как описано выше для Метода ASK, за исключением того, что данная команда может быть адресована множеству радиочастотных меток, и множество радиочастотных меток может отвечать на данную команду.

6.3.3.4.11.3.9 Команда *BlockPermalock* (необязательная)

УСО и радиочастотные метки могут включать в себя средства реализации команды *BlockPermalock*; если это так, то команда должна быть реализована, как приведено в таблице 80. Команда *BlockPermalock* позволяет УСО:

- устанавливать «постоянную блокировку» одному и более блокам (отдельным подсегментам) в Пользовательской памяти радиочастотной метки или

- считывать состояние «постоянной блокировки» блоков памяти в Пользовательской памяти радиочастотной метки.

Одиночная команда *BlockPermalock* может установить «постоянную блокировку» от 0 до 4080 блоков Пользовательской памяти. Размер блоков устанавливается изготовителем. Блоки памяти необязательно должны быть последовательными.

Выполнять команду *BlockPermalock* должны только радиочастотные метки, находящиеся в состоянии **secured**.

Команда *BlockPermalock* отличается от команды *Lock* тем, что команда *BlockPermalock* устанавливает «постоянную блокировку» блоков Пользовательской памяти в незаписываемое состояние, тогда как команда *Lock* устанавливает обратимую или постоянную блокировку на пароль или на весь банк

памяти в записываемое или незаписываемое состояние. 0 определяет, как радиочастотная метка реагирует на команду, *BlockPermalock* (с Read/Lock = 1), следующую за командой *Lock*, и наоборот.

Таблица 79 — Приоритеты для команд *Lock* и *BlockPermalock*

| Первая команда | | | Вторая команда | | | Действие радиочастотной метки и ответ на 2-ю команду |
|---|---|---|---|---|---|---|
| *Lock* | Пароль записи | Постоянная блокировка | *BlockPermalock* (Read/Lock = 1) | | | — |
| | 0 | 0 | | | | Постоянная блокировка блоков, указанных Mask; ответ согласно 6.3.3.4.11.3.9 |
| | 0 | 1 | | | | Команда *BlockPermalock* отклонена; ответ с кодом ошибки |
| | 1 | 0 | | | | Постоянная блокировка блоков, указанных Mask; ответ согласно 6.3.3.4.11.3.9 |
| | 1 | 1 | | | | Постоянная блокировка блоков, указанных Mask; ответ согласно 6.3.3.4.11.3.9 |
| *BlockPermalock* (Read/Lock = 1) | | | *Lock* | Пароль записи | Постоянная блокировка | — |
| | | | | 0 | 0 | Выполнить команду *Lock*, но не снимать постоянную блокировку ни с каких блоков, которые были предварительно постоянно заблокированы; ответ согласно 6.3.3.4.11.3.5 |
| | | | | 0 | 1 | Команда *Lock* отклонена; ответ с кодом ошибки |
| | | | | 1 | 0 | Выполнить команду *Lock*, но не снимать постоянную блокировку ни с каких блоков, которые были предварительно постоянно заблокированы; ответ согласно 6.3.3.4.11.3.5 |
| | | | | 1 | 1 | Выполнить команду *Lock*; ответ согласно 6.3.3.4.11.3.5 |

Команда *BlockPermalock* имеет следующие поля:

- MemBank определяет, к какому блоку памяти команда *BlockPermalock* будет иметь доступ: к Резервному, UII, TID или Пользовательскому. Команда *BlockPermalock* должна применяться только к одному банку памяти. Последующие команды *BlockPermalock* могут применяться к другим банкам. Радиочастотные метки могут выполнить команду *BlockPermalock*, только если MemBank = $11_2$ (Пользовательская память); Если радиочастотная метка принимает команду *BlockPermalock* с MemBank <>$11_2$, то она должна игнорировать команду, передать код ошибки (см. приложение Е) и остаться в состоянии secured.
- Read/Lock определяет, будет ли радиочастотная метка передавать статус выключенной «постоянной блокировки» или «постоянной блокировки» одного или более блоков банка памяти, указанного в MemBank. Радиочастотная метка должна интерпретировать бит Read/Lock следующим образом:
  - Read/Lock = 0: Радиочастотная метка должна передать статус постоянной блокировки блоков указанного банка памяти, начиная с блока памяти, расположенного по адресу BlockPtr и заканчивая блоком памяти, расположенным по адресу BlockPtr+(16×BlockRange)–1. Радиочастотная метка должна передать "0", если блок памяти, соответствующий этому биту, не имеет «постоянной блокировки», и "1" если блок имеет «постоянную блокировку». УСО исключает поле Mask из команды BlockPermalock при значении поля Read/Lock = 0.
  - Read/Lock = 1: Радиочастотная метка должна установить «постоянную блокировку» тем блокам в заданном банке памяти, которые указаны в Mask, начиная с адреса BlockPtr и заканчивая адресом BlockPtr+(16×BlockRange)–1.

- BlockPtrlength, BlockPtr указывают начальный адрес для поля Mask в единицах по 16 блоков. Например, BlockPtr = $00_h$ обозначает блок 0, BlockPtr = $01_h$ обозначает блок 16, а BlockPtr = $02_h$ обозначает блок 32. Поле BlockPtr имеет длину 8, 16, 24 или 32 бита, как определено в WordPtrlength.

- BlockRange указывает диапазон Mask, начиная с BlockPtr и заканчивая (16*BlockRange)–1 последними блоками. Если BlockRange = $00_h$, то радиочастотная метка должна игнорировать команду *BlockPermalock* и вместо этого отправить обратно код ошибки (см. приложение Е), оставаясь в состоянии **secured**.

- Mask указывает, каким блокам памяти радиочастотной метки задается «постоянная блокировка». Параметр Mask зависит от бита Read/Lock следующим образом:

- Read/Lock = 0: УСО должно исключить поле Mask из команды BlockPermalock.

- Read/Lock = 1: УСО должно включить поле Mask длиной 16×BlockRange бит в команду BlockPermalock. Биты поля Mask должны располагаться в блоке от МЗР до самого СЗР (т. е., если BlockPtr = $00_h$ текущий бит Mask относится к блоку 0). Радиочастотная метка должна интерпретировать каждый бит Mask следующим образом.

- Mask = 0: Сохранить текущее состояние «постоянной блокировки» для соответствующего блока памяти.

- Mask = 1: «Постоянная блокировка» соответствующего блока памяти. Если блок уже имеет «постоянную блокировку», то радиочастотная метка должна сохранить текущую настройку «постоянной блокировки». Для блока памяти, однажды переведенного в «постоянную блокировку», не может быть снята «постоянная блокировка», за исключением случая повторного «ввода в действие» радиочастотной метки (см. 6.3.3.4.10).

Следующий пример иллюстрирует использование полей Read/Lock, BlockPtr, BlockRange и Mask: Если Read/Lock =1, BlockPtr=$01_h$ и BlockRange =$01_h$, радиочастотная метка работает с шестнадцатью блоками памяти, начиная с блока 16 и заканчивая блоком 31, устанавливает «постоянную блокировку» тех блоков, чьи соответствующие биты в поле Mask активированы.

Команда *BlockPermalock* содержит 8 битов RFU. УСО должно установить эти биты в $00_h$, когда устанавливает связь с радиочастотной меткой. Если радиочастотная метка принимает команду *BlockPermalock*, содержащую ненулевые биты RFU, то она должна игнорировать команду, ответить кодом ошибки (см. приложение Е) и остаться в состоянии **secured**.

Команда *BlockPermalock* также содержит маркер подлинности радиочастотной метки и CRC-16c. CRC-16 рассчитывается от первого бита кода команды до последнего бита маркера подлинности. Если радиочастотная метка принимает команду *BlockPermalock* с правильным битом CRC-16c, но с неверным маркером подлинности, она должна игнорировать команду *BlockPermalock* и остаться в состоянии **secured**.

Если радиочастотная метка принимает команду *BlockPermalock*, которую не может выполнить, так как Пользовательская память не существует, или для нее установлены МЗР и/или второй МЗР слова XPC радиочастотной метки (см. таблицу 38), или в команде один активированный бит поля Mask ссылается на несуществующий блок памяти, то радиочастотная метка должна игнорировать команду *BlockPermalock* и вместо этого передать код ошибки (см. приложение Е), оставаясь в состоянии **secured**. Радиочастотная метка должна рассматривать команду *BlockPermalock*, в которой Read/Lock =1, но Mask имеет длину, не равную 16×BlockRange бит, как неверную (см. 6.3.3.4.11).

Некоторые радиочастотные метки, в зависимости от исполнения изготовителем, могут быть неспособны выполнять команду *BlockPermalock* с некоторыми значениями полей BlockPtr и BlockRange, в этом случае радиочастотная метка должна игнорировать *BlockPermalock* и вместо этого передавать код ошибки (см. приложение Е), оставаясь в состоянии **secured**. Поскольку радиочастотная метка хранит информацию в своей памяти TID, то УСО может использовать для уникальной идентификации дополнительные возможности, которые поддерживает радиочастотная метка (см. 6.3.3.4.1.3), УСО должно считывать память TID радиочастотной метки перед подачей команды *BlockPermalock*.

Если УСО подает команду *BlockPermalock*, в которой поля BlockPtr и BlockRange указывают один или более несуществующих блоков, но поле Mask активирует «постоянную блокировку» только существующих блоков, то радиочастотная метка должна выполнить эту команду.

Команде *BlockPermalock* должен предшествовать префикс с кадром синхронизации (см. 6.3.3.3.1.2.8).

После подачи команды *BlockPermalock* УСО должно передать CW длительностью наименьшей из значений: $T_{ОТВЕТА}$ или 20 мс, где $T_{ОТВЕТА}$ — это время между подачей УСО команды *BlockPermalock* и передачей ответа радиочастотной меткой модуляцией нагрузки. Время $T_{ОТВЕТА}$ необходимо умножить

на типовое значение $T_1$ (см. таблицу 41) с отклонением ±2,4 мкс, так что $T_{ОТВЕТА} = n*1024/f_c \pm 32/f_c$. УСО может иметь несколько возможных исходов выполнения команды BlockPermalock, зависящие от значения бита Read/Lock в команде и, если Read/Lock = 1, от успеха или неудачи операции блокировки памяти радиочастотной метки:

- **Read/Lock = 0 и радиочастотная метка способна выполнить команду:** радиочастотная метка должна передать ответ, приведенный в таблице 81, в течение интервала T1 в таблице 41, включая заголовок (бит 0), запросить биты «постоянной блокировки», маркер подлинности радиочастотной метки и CRC-16c, рассчитанный от 0 бита «постоянной блокировки» и до маркера подлинности. Ответ радиочастотной метки должен использовать заголовок, определяемый значением TRext в команде *BeginRound*, которая инициировала этот цикл.

- **Read/Lock = 0 и радиочастотная метка не способна выполнить команду:** радиочастотная метка должна выполнить передачу кода ошибки за интервал T1 из таблицы 41; вместо ответа, приведенного в таблице 81 (см. приложение Е для определения кодов ошибки и формата ответа). Ответ радиочастотной метки должен использовать заголовок, определяемый значением TRext в команде *BeginRound*, которая инициировала этот цикл.

- **Read/Lock = 1 и команда *BlockPermalock* выполнена успешно:** После завершения команды *BlockPermalock* радиочастотная метка должна передать ответ, приведенный в таблице 82, включая заголовок (бит 0), маркер подлинности радиочастотной метки и CRC-16c , рассчитанный от 0 бита до маркера подлинности. Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockPermalock* завершена успешно. Ответ радиочастотной метки должен использовать заголовок, указанный в команде *BeginRound*, которая инициировала этот цикл.

- **Read/Lock = 1, и команда *BlockPermalock* не была успешной:** Если УСО получает этот ответ в интервале 20 мс, то считает, что команда *BlockPermalock* не была завершена успешно. УСО может подать команду *Req_RN* (содержащую маркер подлинности радиочастотной метки) для проверки того, что радиочастотная метка по-прежнему находится в поле УСО, или может повторно подать команду *BlockPermalock*.

- **Read/Lock = 1, и радиочастотная метка обнаружила ошибку:** радиочастотная метка должна ответить кодом ошибки в течение интервала CW, вместо ответа, приведенного в таблице 82 (см. приложение Е для определения кодов ошибки и формата ответа). Ответ радиочастотной метки должен использовать заголовок, указанный в команде *BeginRound*, которая инициировала этот цикл. Время $T_{ОТВЕТА}$ необходимо умножить на типовое значение T1 (см. таблицу 41) с отклонением ±2,4 мкс так, что $T_{ОТВЕТА} = n*1024/fc \pm 32/fc$.

Таблица 80 — Команда *BlockPermalock*

| Свойство | Команда | RFU | Read/Lock | MemBank | Length BlockPtr | BlockPtr | Block Range | Mask | RN | CRC-16c |
|---|---|---|---|---|---|---|---|---|---|---|
| Число битов | 8 | 8 | 1 | 2 | 2 | 8, 16, 24 или 32 | 8 | Переменная | 16 | 16 |
| Описание | 11001001 | $00_h$ | 0: считывание<br>1: постоянная блокировка | 00: RFU<br>01: UII<br>10: TID<br>11: Пользователь | Длина указателя в словах<br>00: 8 битов<br>01: 16 битов<br>10: 24 бита<br>11: 32 бита | Mask в единицах по 16 блоков | Mask в единицах по 16 блоков | 0: Сохранить текущую настройку постоянной блокировки<br>1: Активирована постоянная блокировка | Маркер подлинности | — |

Таблица 81 — Ответ радиочастотной метки на успешную команду *BlockPermalock* с битом Read/Lock = 0

| Свойство | Заголовок | Данные | RN | CRC-16c |
|---|---|---|---|---|
| Число битов | 1 | Переменная | 16 | 16 |
| Описание | 0 | Биты постоянной блокировки | Маркер подлинности | — |

Таблица 82 — Ответ радиочастотной метки на успешную команду BlockPermalock с битом Read/Lock = 1

| Свойство | Заголовок | RN | CRC-16c |
|---|---|---|---|
| Число битов | 1 | 16 | 16 |
| Описание | 0 | Маркер подлинности | – |

После приема действительной команды *BlockPermalock* радиочастотная метка должна выполнить предписанную операцию, за исключением случая, когда радиочастотная метка не поддерживает постоянную блокировку блока, в этом случае она должна игнорировать команду.

Метод PJM: См. 6.3.3.4.9 и 6.3.3.4.11. Также для метода *ACK-PJM* см. 6.3.3.4.11.2.4 и таблицу 57, эта команда может содержать несколько маркеров подлинности в поле RN для адресации множества радиочастотных меток. УСО и радиочастотные метки должны работать (при использовании этой команды), как описано выше для Метода ASK, за исключением того, что этой командой может адресоваться множество радиочастотных меток, и множество радиочастотных меток может отвечать на эту команду.

## 7 Маркировка оборудования

Все устройства считывания/опроса (в т. ч. их сопроводительная документация) должны быть четко и без пропусков промаркированы установленным образом, в соответствии с национальными нормативами.

Все устройства считывания/опроса (в т. ч. их сопроводительная документация) должны быть четко и без пропусков промаркированы, чтобы показать, какие МОДА в соответствии с настоящим стандартом они поддерживают.

**Приложение А**
**(справочное)**

## МОДА 1: обязательные и дополнительные команды, требующиеся для поддержки протокола передачи данных по *[3]*

*[3]* является неотъемлемой частью документации РЧИ для систем управления предметами и определяет правила кодирования данных как для открытых, так и для закрытых системных приложений. Для реализации возможности кодирования на радиочастотную метку и считывания с радиочастотной метки данных приложений системам РЧИ необходим ряд дополнительных команд, определенных в *ГОСТ Р ИСО/МЭК 15693-3*, требующихся радиочастотной метке для реализации МОДА 1 в соответствии с настоящим стандартом для выполнения функций, необходимых для управления предметами. Таблица А.1 определяет эти требования и другие условия. Сами команды остаются точно такими, как определено в *ГОСТ Р ИСО/МЭК 15693-3*.

Таблица А.1 — Команды по *ГОСТ Р ИСО/МЭК 15693-3*, требуемые для поддержки протокола передачи данных по *[3]*

| Команда Код | Функция | Статус по *ГОСТ Р ИСО/МЭК 15693-3* | Статус для поддержки протокола передачи данных по *[3]* |
|---|---|---|---|
| $01_h$ | Инвентаризация | Обязательная | Требуется |
| $02_h$ | Состояние покоя | Обязательная | Требуется для радиоинтерфейса |
| $20_h$ | Считывание одиночного блока | Необязательная | Требуется одна из двух команд считывания |
| $23_h$ | Считывание нескольких блоков | Необязательная | — |
| $21_h$ | Запись одиночного блока | Необязательная | Требуется одна из двух команд записи |
| $24_h$ | Запись нескольких блоков | Необязательная | — |
| $22_h$ | Блокировка блока | Необязательная | Требуется |
| $25_h$ | Выбор | Необязательная | Требуется для радиоинтерфейса |
| $26_h$ | Сбросить в состояние ready | Необязательная | Требуется для радиоинтерфейса |
| $27_h$ | Запись AFI | Необязательная | Требуется |
| $28_h$ | Блокировка AFI | Необязательная | Желательна для полной поддержки |
| $29_h$ | Запись DSFID | Необязательная | Желательна для полной поддержки (см. примечание) |
| $2A_h$ | Блокировка DSFID | Необязательная | Желательна для полной поддержки |
| $2B_h$ | Получение системной информации | Необязательная | Требуется |
| $2C_h$ | Получение статуса защиты нескольких блоков | Необязательная | Требуется |

Примечание — Поддержка для DSFID требуется в [3]. Для радиочастотных меток МОДА 1 в соответствии с настоящим стандартом не поддерживается команда радиоинтерфейса «Запись DSFID», [3] определяет метод для кодирования этого параметра в Пользовательской памяти.

**Приложение В**
**(справочное)**

## МОДА 2 и МОДА 3: фазовая модуляция колебаний

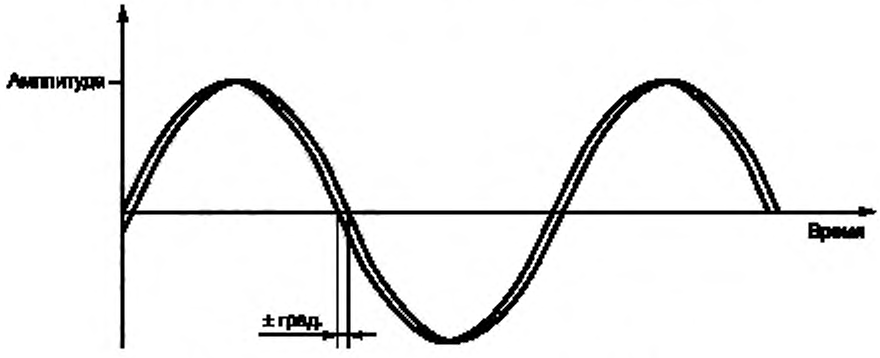



Рисунок В.1 — Фазовая модуляция колебаний (PJM)

PJM состоит из двух компонентов:

1) Синфазный (0°) мощный сигнал I.
2) Низкоуровневый квадратурный (90°) информационный сигнал ±Q.

Форма сигнала PJM является суммой этих двух сигналов. Используя фазовые векторы, их можно представить, как показано на рисунке В.2.

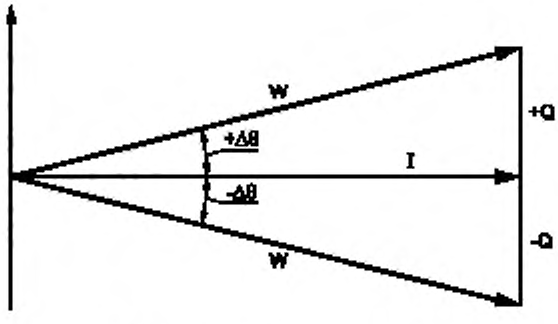



Рисунок В.2 — Частотный спектр

Частотный спектр компонентов фазовых векторов показан на рисунке В.3:

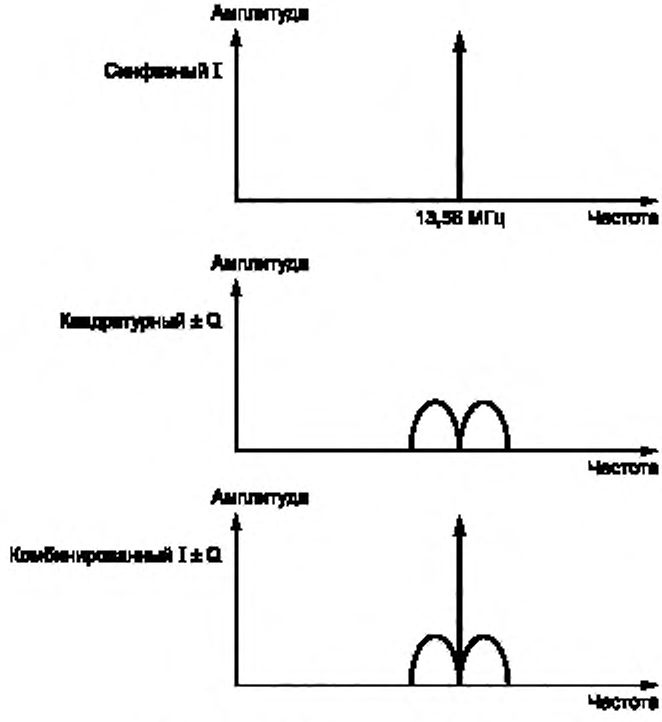



Рисунок В.3 — Формирование PJM

В МОДА 2 для настоящего стандарта:

- скорость передачи данных в битах УСО 423,75 Кбит/с, см. M2-Int:9;
- детали изменения фазы определены в M2-Int: 7 и 6.2.3.2.1.

В МОДА 3 для настоящего стандарта:

- скорость передачи данных в битах УСО 212 Кбит/с, см. M2-Int:9;
- детали изменения фазы определены в M3-Int: 7 и 6.3.3.3.1.2.5.

Особенности PJM:

- постоянная амплитуда сигнала при постоянной энергии передачи;
- уровни боковых полос частот не зависят от скорости передачи данных и могут настраиваться для удовлетворения требований нормативных актов;
- можно передавать на очень высокой скорости передачи данных, так как полоса пропускания Метода PJM не шире основной двусторонней полосы пропускания передачи данных;
- узкополосные антенны не ограничивают скорость передачи сигналов PJM. PJM может быть предварительно скомпенсированным для отмены влияния воздействия ширины полосы пропускания антенны.

Пример реализации PJM приведен на рисунке В.4 и на рисунке В.5:

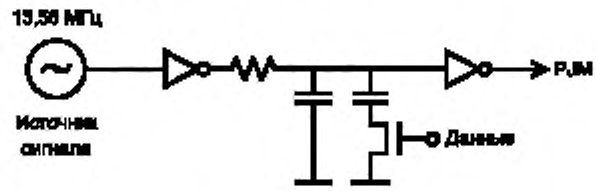



Рисунок В.4 — Пример реализации — простая цепь для обеспечения управляемого данными переменного сдвига фазы для генерации PJM

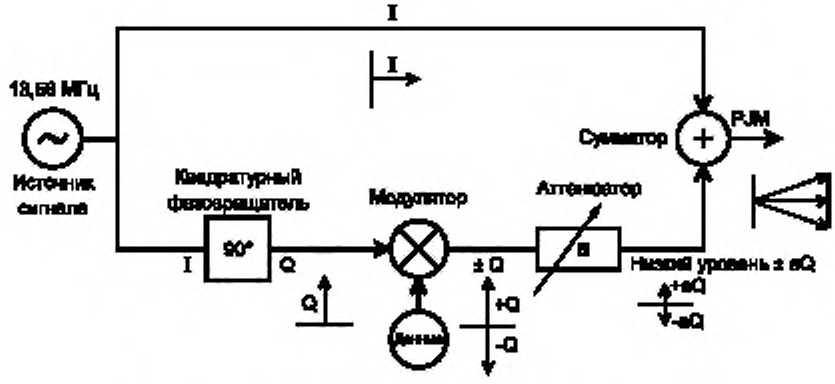



Рисунок В.5 — Пример реализации — цепь для генерации PJM, показывающая различные элементы сигнала PJM

**Приложение С**
**(обязательное)**

## МОДА 3: таблицы перехода состояний

В таблицах перехода состояний от таблицы С.1 до таблицы С.7 определены ответы радиочастотной метки на команды УСО. Термин маркер подлинности, используемый в таблицах перехода состояний, определен в разделе 6.3.3.4.4; коды ошибки определены в таблице Е.2; «слот» — это выходное значение счетчика слота, показанного на рисунке 43 и подробно описанного в приложении F; знак «–» в столбце «Действие» означает, что радиочастотная метка не изменяет свои флаги **SL** и **inventoried**, а также не передает ответ.

### С.1 Текущее состояние: ready

Таблица С.1 — Таблица переходов для состояния ready

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечание 1) | слот=0; соответствие флагов **inventoried** и **SL** | Передача кода StoredCRC | **reply** |
| | слот<>0; соответствие флагов **inventoried** и **SL** | — | **arbitrate** |
| | Иначе | — | **ready** |
| *NextSlot* | Все | — | **ready** |
| *ResizeRound* | Все | — | **ready** |
| *ACK* | Все | — | **ready** |
| *NAK* | Все | — | **ready** |
| *Req_RN* | Все | — | **ready** |
| *Read* | Все | — | **ready** |
| *Write* | Все | — | **ready** |
| *Kill* | Все | — | **ready** |
| *Lock* | Все | — | **ready** |
| *Access* | Все | — | **ready** |
| *BlockWrite* | Все | — | **ready** |
| *BlockErase* | Все | — | **ready** |
| *BlockPermalock* | Все | — | **ready** |
| Недопустимая команда (примечание 2) | Все | — | **ready** |

Примечания

1 *BeginRound* запускает новый цикл и может изменить сеанс. *BeginRound* также дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### С.2 Текущее состояние: arbitrate

Таблица С.2 — Таблица переходов для состояния **arbitrate**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечания 1, 2) | Слот=0; соответствие флагов **inventoried** и **SL** | Передача кода StoredCRC | **reply** |
| | Слот<>0; соответствие флагов **inventoried** и **SL** | — | **arbitrate** |
| | Иначе | — | **ready** |
| *NextSlot* | Слот=0 после декрементирования счетчика слота | Передача кода StoredCRC | **reply** |
| | Слот<>0 после декрементирования счетчика слота | — | **arbitrate** |
| *ResizeRound* (примечание 2) | Слот=0 | Передача кода StoredCRC | **reply** |
| | Слот <>0 | — | **arbitrate** |
| *ACK* | Все | — | **arbitrate** |
| *NAK* | Все | — | **arbitrate** |
| *Req_RN* | Все | — | **arbitrate** |
| *Select* | Все | Активирован или нет **SL**, или установка **inventoried** в *A* или *B* | **ready** |
| *Read* | Все | — | **arbitrate** |
| *Write* | Все | — | **arbitrate** |
| *Kill* | Все | — | **arbitrate** |
| *Lock* | Все | — | **arbitrate** |
| *Access* | Все | — | **arbitrate** |
| *BlockWrite* | Все | — | **arbitrate** |
| *BlockErase* | Все | — | **arbitrate** |
| *BlockPermalock* | Все | — | **arbitrate** |
| Недопустимая команда (примечание 3) | Все | — | **arbitrate** |

Примечания

1 *BeginRound* запускает новый цикл и может изменить сеанс.

2 *BeginRound* и *ResizeRound* также дают команду радиочастотной метке на загрузку нового случайного значения в ее счетчик слота.

3 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (отличную от *BeginRound*) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### С.3 Текущее состояние: reply

Таблица С.3 — Таблица переходов для состояния **reply**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечания 1, 2) | Слот=0; соответствие флагов **inventoried** и **SL** | Передача кода StoredCRC | **reply** |
| | Слот<>0; соответствие флагов **inventoried** и **SL** и SL | — | **arbitrate** |
| | Иначе | — | **ready** |
| *NextSlot* | Все | — | **arbitrate** |
| *ResizeRound* (примечание 2) | Слот=0 | Передача кода StoredCRC | **reply** |
| | Слот <>0 | — | **arbitrate** |
| *ACK* | Действительные коды StoredCRC | См. таблицу 42 | **acknowledged** |
| | Неверные коды StoredCRC | — | **arbitrate** |
| *NAK* | Все | — | **arbitrate** |
| *Req_RN* | Все | — | **arbitrate** |
| *Select* | Все | Активирован или нет **SL**, или установка **inventoried** в *A* или *B* | **ready** |
| *Read* | Все | — | **arbitrate** |
| *Write* | Все | — | **arbitrate** |
| *Kill* | Все | — | **arbitrate** |
| *Lock* | Все | — | **arbitrate** |
| *Access* | Все | — | **arbitrate** |
| *BlockWrite* | Все | — | **arbitrate** |
| *BlockErase* | Все | — | **arbitrate** |
| *BlockPermalock* | Все | — | **arbitrate** |
| $T_2$ пауза | См. рисунок 40 и таблицу 41 | — | **arbitrate** |
| Недопустимая команда (примечание 3) | Все | — | **reply** |

Примечания

1 *BeginRound* начинает новый цикл и может изменить сеанс.

2 *BeginRound* и *ResizeRound* также дают команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

3 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (отличную от *BeginRound*) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### С.4 Текущее состояние: acknowledged

Таблица С.4 — Таблица переходов для состояния **acknowledged**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечание 1) | Слот=0; соответствие флагов **inventoried** (примечание 2) и **SL** | Передача кода StoredCRC; переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **reply** |
| | Слот<>0; соответствие флагов **inventoried** (примечание 2) и **SL** | Переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предществующему сеансу | **arbitrate** |
| | Иначе | Переход **inventoried** из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **ready** |
| *NextSlot* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ResizeRound* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ACK* | Действительные коды StoredCRC | См. таблицу 42 | **acknowledged** |
| | Неверные коды StoredCRC | — | **arbitrate** |
| *NAK* | Все | — | **arbitrate** |
| *Req_RN* | Действительные коды StoredCRC и пароль доступа<>0 | Маркер подлинности | **open** |
| | Действительные коды storedCRC и пароль доступа=0 | Маркер подлинности | **secured** |
| | Неверные коды StoredCRC | — | **acknowledged** |
| *Select* | Все | Активирован или нет **SL**, или установка **inventoried** в *A* или *B* | **ready** |
| *Read* | Все | — | **arbitrate** |
| *Write* | Все | — | **arbitrate** |
| *Kill* | Все | — | **arbitrate** |
| *Lock* | Все | — | **arbitrate** |
| *Access* | Все | — | **arbitrate** |
| *BlockWrite* | Все | — | **arbitrate** |
| *BlockErase* | Все | — | **arbitrate** |
| *BlockPermalock* | Все | — | **arbitrate** |
| $T_2$ пауза | См. рисунок 40 и таблицу 41 | — | **arbitrate** |
| Недопустимая команда (примечание 3) | Все | — | **acknowledged** |

Примечания

1 *BeginRound* начинает новый цикл и может изменить сеанс. *BeginRound* также дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 Как описано в 6.3.3.4.8, радиочастотная метка переключает свой флаг **inventoried** до определения условия.

3 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (отличную от *BeginRound*) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### C.5 Текущее состояние: open

Таблица C.5 — Таблица переходов для состояния **open**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечание 1) | Слот=0; соответствие флагов **inventoried** (примечание 2) и **SL** | Передача кода StoredCRC; переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **reply** |
| | Слот<>0; соответствие флагов **inventoried** (примечание 2) и **SL** | Переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **arbitrate** |
| | Иначе | Переход **inventoried** из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **ready** |
| *NextSlot* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ResizeRound* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ACK* | Действительный маркер подлинности | См. таблицу 42 | **open** |
| | Неверный маркер подлинности | — | **arbitrate** |
| *NAK* | Все | — | **arbitrate** |
| *Req_RN* | Действительный маркер подлинности | Передача нового RN16 | **open** |
| | Неверный маркер подлинности | — | **open** |
| *Select* | Все | Активирован или нет **SL**, или установка **inventoried** в *A* или *B* | **ready** |
| *Read* | Действительный маркер подлинности и правильный доступ к памяти | Переданные данные и маркер подлинности | **open** |
| | Действительный маркер подлинности и неверный доступ к памяти | Код ошибки модуляции нагрузки | **open** |
| | Неверный дескриптор | — | **open** |
| *Write* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Действительный маркер подлинности и неверный доступ к памяти | Код ошибки передачи | **open** |
| | Неверный маркер подлинности | — | **open** |

*Окончание таблицы С.5*

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *Kill* (примечание 3) (см. также 6.3.3.4.11.3.4) | Действительный маркер подлинности и правильный ненулевой «пароль уничтожения» и Recom = 0 | Передача маркера подлинности, если выполнено | **killed** |
| | Действительный маркер подлинности и правильный ненулевой пароль уничтожения и Recom <> 0 | Передача маркера подлинности, если выполнено | **open** |
| | Действительный маркер подлинности и неверный ненулевой «пароль уничтожения» | — | **arbitrate** |
| | Действительный маркер подлинности и «пароль уничтожения»=0 | Передача кода ошибки | **open** |
| | Неверный маркер подлинности | — | **open** |
| *Lock* | Все | — | **open** |
| *Access* (см. также рисунок 49) | Действительный маркер подлинности и неверный «пароль доступа» | Передача маркера подлинности | **secured** |
| | Действительный маркер подлинности и неверный «пароль доступа» | — | **arbitrate** |
| | Неверный маркер подлинности | — | **open** |
| *BlockWrite* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Неверный маркер подлинности | — | **open** |
| *BlockErase* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Неверный маркер подлинности | — | **open** |
| *BlockPermalock* | Все | — | **open** |
| Недопустимая команда (примечание 4) | Все, за исключением действительных команд, вставленных в промежутки между поочередно следующими командами *Kill* или *Access* в последовательности «уничтожения» или доступа соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **open** |
| | В противном случае действительные команды, за исключением команд *Req_RN* или *BeginRound*, вставленных в промежутки между поочередно следующими командами *Kill* или *Access* в последовательности «уничтожения» или доступа соответственно (см. рисунок 47 и рисунок 49) | — | arbitrate |

Примечания

1 *BeginRound* начинает новый цикл и может изменить сеанс. *BeginRound* также дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 Как описано в 6.3.3.4.8 (также для метода *ACK-PJM* см. 6.3.3.4.11.2.4), радиочастотная метка переключает свой флаг **inventoried** до определения условия.

3 Как описано в 6.3.3.4.11.3.4, если радиочастотная метка не поддерживает M3P и второй M3P параметра Recom, она игнорирует их значения и обрабатывает их, как будто они нулевые.

4 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (отличную от *BeginRound*) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### С.6 Текущее состояние: secured

Таблица С.6 — Таблица переходов для состояния **secured**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* (примечание 1) | Слот=0; соответствие флагов **inventoried** (примечание 2) и **SL** | Передача кода StoredCRC; переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **reply** |
| | Слот<>0: соответствие флагов **inventoried** (примечание 2) и **SL** | Переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **arbitrate** |
| | Иначе | Переход **inventoried** из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **ready** |
| *NextSlot* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ResizeRound* | Все | Переход **inventoried** от *A*→*B* | **ready** |
| *ACK* | Действительный маркер подлинности | См. таблицу 42 | **secured** |
| | Неверный маркер подлинности | — | **arbitrate** |
| *NAK* | Все | — | **arbitrate** |
| *Req_RN* | Действительный маркер подлинности | Передача нового RN16 | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| *Select* | Все | Активирован или нет **SL**, или установка **inventoried** в *A* или *B* | **ready** |
| *Read* | Действительный маркер подлинности и правильный доступ к памяти | Передача данных и маркера подлинности | **secured** |
| | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Неверный дескриптор | — | **secured** |
| *Write* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности | — | **secured** |

*Продолжение таблицы С.6*

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *Kill* (примечание 3) (см. 6.3.3.4.11.3.4) | Действительный маркер подлинности и правильный ненулевой пароль уничтожения и Recom = 0 | Передача маркера подлинности, если выполнено | **killed** |
| | Действительный маркер подлинности и правильный ненулевой пароль уничтожения и Recom <> 0 | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности и неверный ненулевой пароль уничтожения | — | **arbitrate** |
| | Действительный маркер подлинности и пароль уничтожения=0 | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| *Lock* | Действительный маркер подлинности и действительная полезная информация блокировки | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности и неверная полезная информация блокировки | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| *Access* (см. также рисунок 49) | Действительный маркер подлинности и неверный пароль доступа | Передача маркера подлинности | **secured** |
| | Действительный маркер подлинности и неверный пароль доступа | — | **arbitrate** |
| | Неверный маркер подлинности | — | **secured** |
| *BlockWrite* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности и неверный доступ к памяти | Код ошибки передачи | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| *BlockErase* | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности р | — | **secured** |
| *BlockPermalock* | Действительный маркер подлинности, правильная строка параметров, и Read/Lock = 0 | Передача битов «постоянной блокировки» и маркера подлинности | **secured** |
| | Действительный маркер подлинности, неверная строка параметров, и Read/Lock = 0 | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности, правильная строка параметров, и Read/Lock = 1 | Передача маркера подлинности, если выполнено | **secured** |

*Окончание таблицы С.6*

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BlockPermalock* | Действительный маркер подлинности, неверная строка параметров, и Read/Lock = 1 | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| Недопустимая команда (примечание 4) | Все, за исключением действительных команд, вставленных в промежутки между поочередно следующими командами *Kill* или *Access* в последовательности *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **secured** |
| | В противном случае действительные команды, за исключением команд *Req_RN* или *BeginRound*, вставленных в промежутки между поочередно следующими командами *Kill* или *Access* в последовательности *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **arbitrate** |

Примечания

1 *BeginRound* начинает новый цикл и может изменить сеанс. *BeginRound* также дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 Как описано в 6.3.3.4.8, радиочастотная метка переключает свой флаг **inventoried** до определения условия.

3 Как описано в 6.3.3.4.11.3.4, если радиочастотная метка не поддерживает M3P и второй M3P параметра Recom, она игнорирует их значения и обрабатывает их, как будто они нулевые.

4 Запись «Недопустимая команда» должна означать несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (отличную от *BeginRound*) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

### С.7 Текущее состояние: killed (не обязательно)

Таблица С.7 – Таблица переходов для состояния **killed**

| Команда | Условие | Действие | Следующее состояние |
|---|---|---|---|
| *BeginRound* | Все | — | **killed** |
| *NextSlot* | Все | — | **killed** |
| *ResizeRound* | Все | — | **killed** |
| *ACK* | Все | — | **killed** |
| *NAK* | Все | — | **killed** |
| *Req_RN* | Все | — | **killed** |
| *Select* | Все | — | **killed** |
| *Read* | Все | — | **killed** |
| *Write* | Все | — | **killed** |
| *Kill* | Все | — | **killed** |
| *Lock* | Все | — | **killed** |
| *Access* | Все | — | **killed** |
| *BlockWrite* | Все | — | **killed** |
| *BlockErase* | Все | — | **killed** |
| *BlockPermalock* | Все | — | **killed** |
| Недопустимая команда | Все | — | **killed** |

Примечание — Запись «Недопустимая команда» должна означать неправильную команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, или любую другую команду, не опознанную, либо не выполняемую радиочастотной меткой.

**Приложение D**
**(обязательное)**

## МОДА 3: таблицы ответов на команды

В таблицах от таблицы D.1 до таблицы D.17 определены ответы радиочастотных меток на команды УСО. Термин «маркер подлинности», используемый в таблицах перехода состояний, определен в 6.3.3.4.4, коды ошибки определены в таблице E.2; "слот" — это выходное значение счетчика слота, показанного на рисунке 43 и подробно описанного в приложении F; знак "—" в колонке "Ответ" означает, что радиочастотная метка не изменяет свои флаги **SL**, и **inventoried**, а также не передает ответ.

### D.1 Ответ на команду: Power-up

Таблица D.1 — Таблица ответа на команду Power-up

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready**, **arbitrate**, **reply**, **acknowledged**, **open**, **secured** | Включение питания | — | **ready** |
| **killed** | Все | — | **killed** |

### D.2 Ответ на команду: *BeginRound*

Таблица D.2 — Таблица ответа на команду *BeginRound* (примечание 1)

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready**, **arbitrate**, **reply** | Слот=0; соответствие флагов **inventoried** и **SL** | Передача кода StoredCRC | **reply** |
| | Слот<>0; соответствие флагов **inventoried** и **SL** | — | **arbitrate** |
| | Иначе | — | **ready** |
| **acknowledged**, **open**, **secured** | Слот=0; соответствие флагов **inventoried** (примечание 2) и **SL** | Передача кода StoredCRC; переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **reply** |
| | Слот<>0; соответствие флагов **inventoried** (примечание 2) и **SL** | Переход **inventoried** (примечание 2) из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **arbitrate** |
| | Иначе | Переход к **inventoried** из *A*→*B*, если и только если новый сеанс соответствует предшествующему сеансу | **ready** |
| **killed** | Все | — | **killed** |

Примечания

1 *BeginRound* (в любом состоянии, отличном от **killed**) запускает новый цикл и может изменить сеанс. *BeginRound* также дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 Как описано в 6.3.3.4.8, радиочастотная метка переключает свой флаг **inventoried** до определения условия.

### D.3 Ответ на команду: *NextSlot*

Таблица D.3 — Таблица ответа на команду *NextSlot* (примечание)

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |

*Окончание таблицы D.3*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **arbitrate** | Слот<>0 после декремента счетчика слота | — | **arbitrate** |
| | Слот=0 после декремента счетчика слота | Передача кода StoredCRC | **reply** |
| **reply** | Все | — | **arbitrate** |
| **acknowledged, open, secured** | Все | Переход **inventoried** из *A→B* | **ready** |
| **killed** | Все | — | **killed** |

Примечание — См. таблицу D.17 для ответа радиочастотной метки на команду *NextSlot*, чей параметр сеанса не соответствует текущему циклу инвентаризации.

### D.4 Ответ на команду: *ResizeRound*

Таблица D.4 — Таблица (примечание 2) ответа на команду *ResizeRound* (примечание 1)

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply** | Слот <>0 | — | **arbitrate** |
| | Слот=0 | Передача кода StoredCRC | **reply** |
| **acknowledged, open, secured** | Все | Переход **inventoried** из *A→B* | **ready** |
| **killed** | Все | — | **killed** |

Примечания

1 *ResizeRound*, в состояниях **arbitrate** или **reply**, дает команду радиочастотной метке на загрузку нового случайного значения в его счетчик слота.

2 См. таблицу D.17 для ответа радиочастотной метки на команду *NextSlot*, чей параметр сеанса не соответствует текущему циклу инвентаризации.

### D.5 Ответ на команду: *ACK*

Таблица D.5 — Таблица ответа на команду *ACK*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate** | Все | — | **arbitrate** |
| **reply** | Действительный код StoredCRC | См. таблицу 42 | **acknowledged** |
| | Неверный код StoredCRC | — | **arbitrate** |
| **acknowledged** | Действительный код StoredCRC | См. таблицу 42 | **akcnowledged** |
| | Неверный код StoredCRC | — | **arbitrate** |
| **open** | Действительный маркер подлинности | См. таблицу 42 | **open** |
| | Неверный маркер подлинности | — | **arbitrate** |
| **secured** | Действительный маркер подлинности | См. таблицу 42 | **secured** |
| | Неверный маркер подлинности | — | **arbitrate** |
| **killed** | Все | — | **arbitrate** |

### D.6 Ответ на команду: *NAK*

Таблица D.6 — Таблица ответа на команду *NAK*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged, open, secured** | Все | — | **arbitrate** |
| **killed** | Все | — | **killed** |

### D.7 Ответ на команду: *Req_RN*

Таблица D.7 — Таблица ответа на команду *Req_RN*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply** | Все | — | **arbitrate** |
| **acknowledged** | Действительный код StoredCRC и «пароль доступа» <>0 | Передача маркера подлинности | **open** |
| | Действительный код StoredCRC и «пароль доступа»=0 | Передача маркера подлинности | **killed** |
| | Неверные коды StoredCRC | — | **acknowledged** |
| **open** | Действительный маркер подлинности | Новая передача RN16 | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности | Новая передача RN16 | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.8 Ответ на команду: *Select*

Таблица D.8 — Таблица ответа на команду *Select*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready, arbitrate, ready, acknowledged, open, secured** | Все | Активирует или деактивирует флаг **SL** или устанавливает флаг **inventoried** в *A* или *B* | **ready** |
| **killed** | Все | — | **secured** |

### D.9 Ответ на команду: *Read*

Таблица D.9 — Таблица ответа на команду *Read*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |

*Окончание таблицы D.9*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **open** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача данных и маркера подлинности | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача данных и маркера подлинности | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **Killed** | Все | — | **killed** |

### D.10 Ответ на команду: *Write*

Таблица D.10 — Таблица ответа на команду *Write*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitre, reply, acknowledge** | Все | — | **arbitrate** |
| **open** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.11 Ответ на команду: *Kill*

Таблица D.11 — Таблица ответа на команду *Kill* (примечание 1)

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** (примечание 2) | Действительный маркер подлинности и «пароль уничтожения»=0 | Передача кода ошибки | **open** |
| | Действительный маркер подлинности и неверный ненулевой «пароль уничтожения» | — | **arbitrate** |

*Окончание таблицы D.11*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **open** (примечание 2) | Действительный маркер подлинности и правильный ненулевой «пароль уничтожения» и Recom=0 | Передача маркера подлинности, если выполнено | **killed** |
| | Действительный маркер подлинности и правильный ненулевой «пароль уничтожения» и Recom <>0 | Передача маркера подлинности, если выполнено | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** (примечание 2) | Действительный маркер подлинности и «пароль уничтожения»=0 | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и неверный ненулевой «пароль уничтожения» | — | **arbitrate** |
| | Действительный маркер подлинности и правильный ненулевой «пароль уничтожения» и Recom =0 | Передача маркера подлинности, если выполнено | **killed** |
| | Действительный маркер подлинности и правильный ненулевой «пароль уничтожения» и Recom <>0 | Передача маркера подлинности, если выполнено | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

Примечания

1 См. также рисунок 47.

2 Как описано в 6.3.3.4.11.3.4, если радиочастотная метка не поддерживает МЗР и второй МЗР поля Recom, она игнорирует их значения и обрабатывает их, как будто они нулевые.

### D.12 Ответ на команду: *Lock*

Таблица D.12 — Таблица ответа на команду *Lock*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** | Все | — | **open** |
| **secured** | Действительный маркер подлинности и неверная строка параметров блокировки | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и правильная строка параметров блокировки | Передача маркера подлинности, если выполнено | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.13 Ответ на команду: *Access*

Таблица D.13 — Таблица ответа на команду *Access**

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** | Действительный маркер подлинности и неверный пароль доступа | — | **arbitrate** |
| | Действительный маркер подлинности и неверный пароль доступа | Передача маркера подлинности | **secured** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности и неверный пароль доступа | — | **arbitrate** |
| | Действительный маркер подлинности и неверный пароль доступа | Передача маркера подлинности | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.14 Ответ на команду: *BlockWrite*

Таблица D.14 — Таблица ответа на команду *BlockWrite*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.15 Ответ на команду: *BlockErase*

Таблица D.15 — Таблица ответа на команду *BlockErase*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |

* *См. также рисунок 49.*

*Окончание таблицы D.15*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **open** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **open** |
| | Неверный маркер подлинности | — | **open** |
| **secured** | Действительный маркер подлинности и неверный доступ к памяти | Передача кода ошибки | **secured** |
| | Действительный маркер подлинности и правильный доступ к памяти | Передача маркера подлинности, если выполнено | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.16 Ответ на команду: *BlockPermalock*

Таблица D.16 — Таблица ответа на команду *BlockPermalock*

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate, reply, acknowledged** | Все | — | **arbitrate** |
| **open** | Все | — | **open** |
| **secured** | Действительный маркер подлинности, правильная строка параметров, и **Read/Lock** = 0 | Передача бита «постоянной блокировки» и маркера подлинности | **secured** |
| | Действительный дескриптор, неверная строка параметров, и **Read/Lock** = 0 | код ошибки передачи | **secured** |
| | Действительный маркер подлинности, правильная строка параметров, и **Read/Lock** = 1 | Передача маркера подлинности, если выполнено | **secured** |
| | Действительный маркер подлинности, неверная строка параметров, и **Read/Lock** = 1 | Передача кода ошибки | **secured** |
| | Неверный маркер подлинности | — | **secured** |
| **killed** | Все | — | **killed** |

### D.17 Ответ на команду: $T_2$ timeout

Таблица D.17 — Таблица ответа на команду $T_2$ timeout

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** | Все | — | **ready** |
| **arbitrate** | Все | — | **arbitrate** |
| **reply, acknowledged** | См. рисунок 40 и таблицу 41 | — | **arbitrate** |
| **open** | Все | — | **open** |
| **secured** | Все | — | **secured** |
| **killed** | Все | — | **killed** |

### D.18 Ответ на команду: Недопустимая команда

Таблица D.18 — Таблица ответа на недопустимую команду

| Начальное состояние | Условие | Ответ | Следующее состояние |
|---|---|---|---|
| **ready** (примечание 1) | Все | — | **ready** |
| **arbitrate** (примечание 2) | Все | — | **arbitrate** |
| **reply** (примечание 2) | Все | — | **reply** |
| **acknowledged** (примечание 2) | Все | — | **acknowledged** |
| **open** (примечание 2) | Все, за исключением действительных команд, вставленных в последовательность команд *Kill* или *Access* в процедуре *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **open** |
| | В противном случае действительные команды, за исключением команд *Req_RN* или *BeginRound*, вставленных в последовательность команд *Kill* или *Access* в процедуре *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **arbitrate** |
| **secured** (примечание 2) | Все, за исключением действительных команд, вставленных в последовательность команд *Kill* или *Access* в процедуре *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **secured** |
| | В противном случае действительные команды, за исключением команд *Req_RN* или *BeginRound*, вставленных в последовательность команд *Kill* или *Access* в процедуре *Kill* или *Access* соответственно (см. 6.3.3.4.11.3.4 и 6.3.3.4.11.3.6) | — | **arbitrate** |
| **killed** (примечание 1) | Все | — | **killed** |

Примечания

1 Понятие «Недопустимая команда» означает несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

2 Понятие «Недопустимая команда» означает несоответствующую команду, неподдерживаемую команду, команду с неверными параметрами, команду с ошибкой CRC, команду (не BeginRound) с параметром сеанса, не соответствующим текущему выполняемому циклу инвентаризации, или любую другую команду, либо не опознанную, либо не выполняемую радиочастотной меткой.

**Приложение Е**
**(обязательное)**

## МОДА 3: коды ошибок

Если при выполнении команды доступа, выполняющей считывание из памяти или запись в память, радиочастотная метка обнаружила ошибку и если эта команда содержит маркер подлинности (т. е. команда *Read*, *Write*, *Kill*, *Lock*, *BlockWrite*, *BlockErase* или *BlockPermalock*), то радиочастотная метка должна выполнить передачу кода ошибки, как показано в таблице Е.1.

Если радиочастотная метка поддерживает специфицированные коды ошибок, она должна использовать специфицированные коды ошибок, приведенные в таблице Е.2.

Если радиочастотная метка не поддерживает специфицированные коды ошибок, она должна выполнить передачу кода ошибки $00001111_2$ (указывает на неспецифицированный код ошибки), приведенного в таблице Е.2.

Радиочастотная метка должна выполнять передачу кодов ошибки только в состояниях **open** или **secured**.

Радиочастотная метка не должна выполнять передачу кода ошибки, если она приняла неверную команду доступа; вместо этого она должна игнорировать эту команду.

Если ошибка описывается более чем одним кодом ошибки, преимущество должен иметь более специфицированный код ошибки, он и должен быть передан радиочастотной меткой.

Для кода ошибки в качестве заголовка используется бит 1 в отличие от заголовка нормального ответа радиочастотной метки, для которого используется бит 0.

Таблица Е.1 — Формат ответа радиочастотной метки об ошибке

| Свойство | Заголовок | Код ошибки | RN | CRC-16c |
|---|---|---|---|---|
| Число битов | 1 | 8 | 16 | 16 |
| Описание | 1 | Код ошибки | Маркер подлинности | — |

Таблица Е.2 — Коды ошибки радиочастотной метки

| Поддерживаемый код ошибки | Код ошибки | Название кода ошибки | Описание ошибки |
|---|---|---|---|
| Специфицированные ошибки | $00000000_2$ | Прочие ошибки | «Ловушка» для ошибок, не покрываемых другими кодами |
| | $00000011_2$ | Выход за пределы памяти | Указанная ячейка памяти не существует или поле длины UII не поддерживается радиочастотной меткой |
| | $00000100_2$ | Память заблокирована | Указанная ячейка памяти заблокирована и/или «постоянно заблокирована» и является незаписываемой или нечитаемой |
| | $00001011_2$ | Недостаточное питание | Радиочастотной метке недостаточно питания для выполнения операции записи в память |
| Неспецифицированные | $00001111_2$ | Неспецифицированная ошибка | Радиочастотная метка не поддерживает коды специфицированных ошибок |

**Приложение F**
**(обязательное)**

## МОДА 3: счетчик слота

### F.1 Работа счетчика слота

Как описано в 6.3.3.4.8, радиочастотные метки содержат средства реализации 15-битового счетчика слота. Как описано в 6.3.3.4.9, УСО используют счетчик слота для управления вероятностью ответа радиочастотной метки на команду *BeginRound*, *ResizeRound* или *NextSlot*. После приема команды *BeginRound* или *ResizeRound* радиочастотная метка предварительно загружает *Q*-битовое значение из RNG радиочастотной метки (см. 6.3.3.4.8) в свой счетчик слота. Значение *Q* — целое число в диапазоне (0,15). Команда *BeginRound* определяет *Q*; команда *ResizeRound* может изменить *Q* из предшествующей ей команды *BeginRound*. После приема команды *NextSlot* радиочастотная метка декрементирует это значение слота. Радиочастотные метки переходят в состояние **reply**, когда их значение слота достигает $0000_h$. Счетчик слота выполняет непрерывный счет; это означает, что после уменьшения значения слота до $0000_h$ следующая команда *NextSlot* вызовет переполнение и счет начнется в режиме вычитания со значения $7FFF_h$. Радиочастотные метки, которые вернулись в состояние **arbitrate** (например, из состояния **reply**) при значении слота $0000_h$, должны декрементировать свой счетчик слота из $0000_h$ до $7FFF_h$ по следующей команде *NextSlot* (в соответствии со значением сеанса) и, поскольку их значение слота теперь не нулевое, должны оставаться в **arbitrate**.

Приложение C и приложение D содержат таблицы, описывающие ответ радиочастотной метки на команды УСО; "слот" в этих таблицах является параметром.

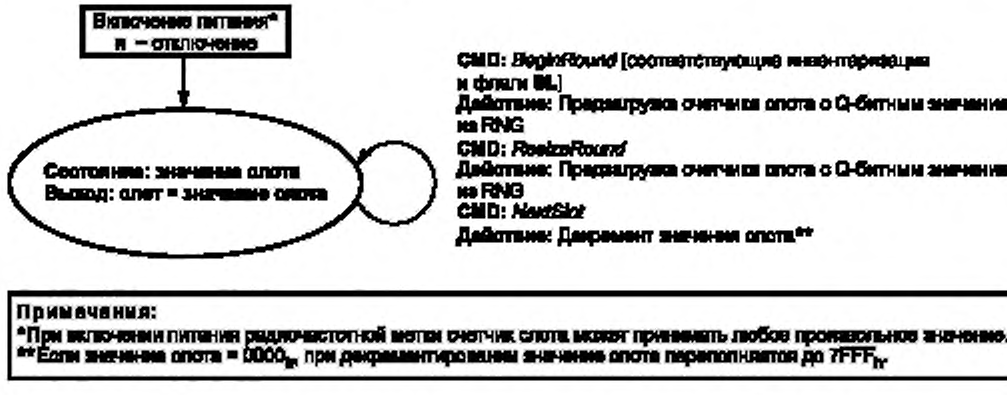



Рисунок F.1 — Диаграмма состояния счетчика слота

**Приложение G**
**(справочное)**

## МОДА 3: пример алгоритма выбора параметра *Q* счетчика слота

### G.1 Пример алгоритма

На рисунке G.1 показан алгоритм того, как можно использовать УСО для задания параметра счета слота *Q* в команде *BeginRound*. $Q_{fp}$ — это значение *Q* в формате с плавающей запятой; УСО округляет $Q_{fp}$ до целого значения и заменяет им целое значение *Q* в команде *BeginRound*. Типовое значение *C* лежит в диапазоне $0,1 < C < 0,5$. УСО обычно использует меньшее значение *C*, когда *Q* велико, и большее значение *C*, когда *Q* мало.

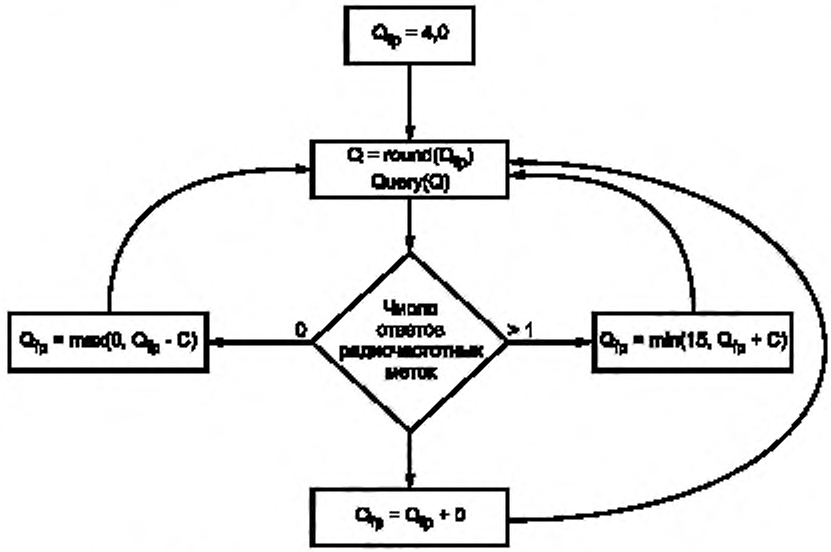


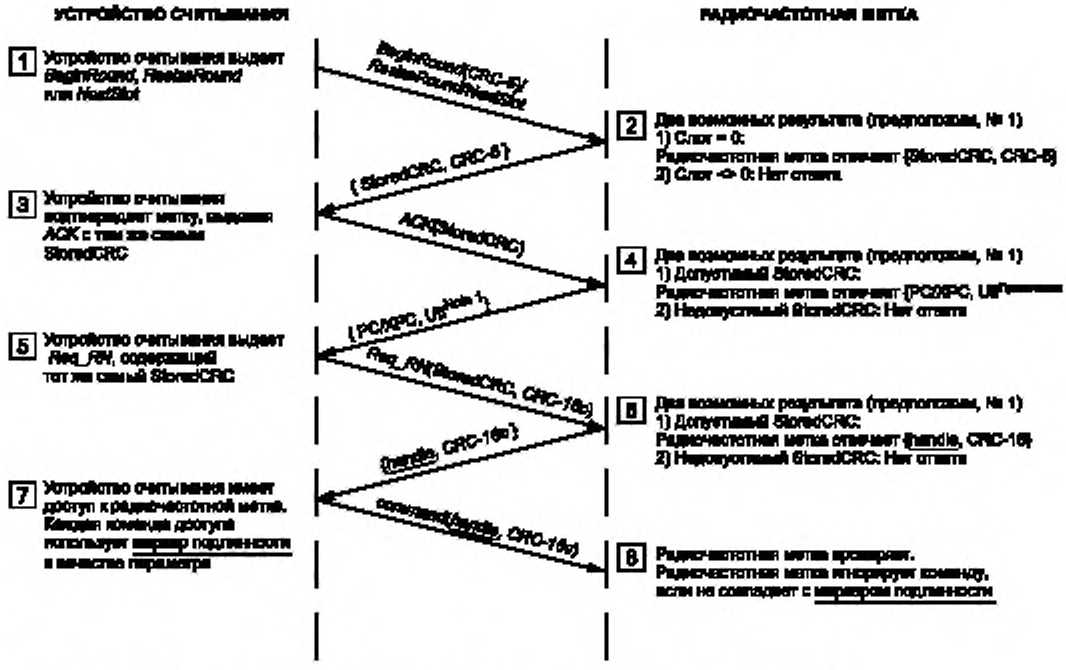

Рисунок G.1 — Пример алгоритма выбора параметра *Q* счетчика слота

Приложение Н
(справочное)

## МОДА 3: пример инвентаризации и доступа к радиочастотной метке

### Н.1 Метод ASK: Пример инвентаризации и доступа к одиночной радиочастотной метке

На рисунке Н.1 показаны шаги, по которым УСО выполняет инвентаризацию и доступ к одиночной радиочастотной метке для Метода ASK.

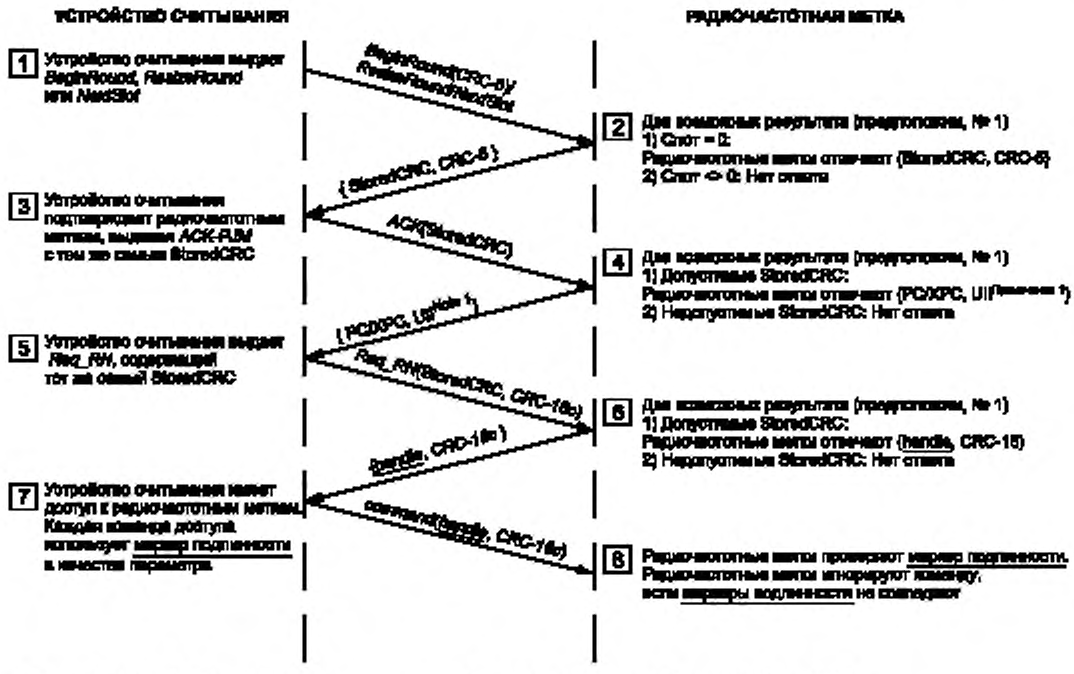



**Примечание 1– В некоторых случаях при необходимости должен добавляться Пакет CRC. Определение ситуаций см. в таблице 42.**

Рисунок Н.1 — Метод ASK: Пример инвентаризации и доступа к радиочастотной метке

### Н.2 Метод PJM: Пример инвентаризации и доступа к одиночной радиочастотной метке или к нескольким радиочастотным меткам

На рисунке Н.2 показаны шаги, по которым УСО выполняет инвентаризацию и доступ к одиночной радиочастотной метке или к нескольким радиочастотным меткам для Метода PJM.




**Примечание 1– В некоторых ситуациях при необходимости должен добавляться Пакет CRC. Определение ситуаций см. в таблице 42.**

Рисунок Н.2 — Метод PJM: Пример инвентаризации и доступа к радиочастотной метке

**Приложение I**
**(справочное)**

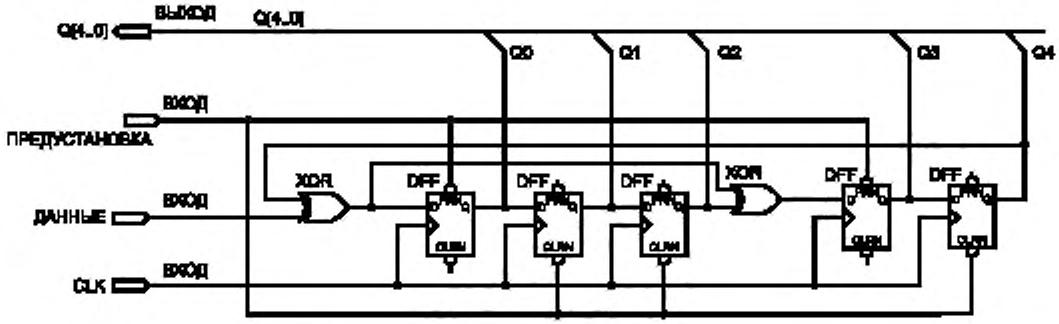
## МОДА 3: расчет 5-битового и 16-битового циклического избыточного кода

### I.1 Пример реализации кодека CRC-5

Пример типовой принципиальной схемы кодека CRC-5 показан на рисунке I.1.

Для кодирования CRC-5 предварительно заносят в регистр CRC (т. е., Q[4:0]) значение $01001_2$, затем подают кодируемые биты данных на вход ДАННЫЕ вперед СЗР по тактам CLK. После ввода всех битов данных регистр Q[4:0] содержит значение CRC-5.

Для декодирования CRC-5 предварительно заносят в регистр CRC (Q4:0]) значение $01001_2$, затем подают принятые данные и код CRC-5, подлежащие декодированию {данные, CRC-5} на вход ДАННЫЕ вперед СЗР. Контроль CRC-5 пройден, если после ввода данных Q[4:0] = $00000_2$.




Риснок I.2 — Пример схемы кодека CRC-5

Таблица I.1 — Начальное значение регистра CRC-5

| Регистр | Предварительно установленное значение |
|---|---|
| Q0 | 1 |
| Q1 | 0 |
| Q2 | 0 |
| Q3 | 1 |
| Q4 | 0 |

### I.2 Пример расчета CRC-16

Здесь приведен пример CRC-16, который радиочастотная метка должна вычислять при включении питания. Как показано на рисунке 41, Память UII содержит коды StoredCRC начиная с адреса $00_h$, данные StoredPC начинаются с адреса $10_h$, ноль или больше слов UII начиная с адреса $20_h$, слово XPC_W1 начиная с адреса $210_h$, и опциональное слово XPC_W2 начинается с адреса $220_h$. Как описано в 6.3.3.4.1.2.1, радиочастотная метка рассчитывает свой код StoredCRC по ее данным StoredPC и битам UII, исключая слова XPC_W1 и XPC_W2 из расчетов. Рисунок 40 приведен для показа примера значений данных StoredPC и слов UII*. В каждом последующем столбце добавлено еще одно слово памяти UII, всей памяти UII, записанной в самом правом столбце. Показанные значения данных StoredPC соответствуют числу записанных слов UII, с битами данных StoredPC $15_h$–$1F_h$, установленными в ноль. Входные данные, помеченные «N/A», означают, что это слово памяти UII не участвует в расчете CRC.

* *Очевидно, в оригинале стандарта ИСО/МЭК 18000-3 имелась в виду таблица I.2.*

Таблица I.2 — Пример содержания памяти UII радиочастотной метки

| Начальный адрес слова UII | Содержимое слова UII | Значения слов UII | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $00_h$ | CRC-16 | $E2F0_h$ | $CCAE_h$ | $968F_h$ | $78F6_h$ | $C241_h$ | $2A91_h$ | $1835_h$ |
| $10_h$ | PC | $0000_h$ | $0800_h$ | $1000_h$ | $1800_h$ | $2000_h$ | $2800_h$ | $3000_h$ |
| $20_h$ | Слово 1 UII | N/A. | $1111_h$ | 1111 | $1111_h$ | $1111_h$ | $1111_h$ | $1111_h$ |
| $30_h$ | Слово 2 UII | N/A | N/A | $2222_h$ | $2222_h$ | $2222_h$ | $2222_h$ | $2222_h$ |
| $40_h$ | Слово 3 UII | N/A | N/A | N/A | $3333_h$ | $3333_h$ | $3333_h$ | $3333_h$ |
| $50_h$ | Слово 4 UII | N/A | N/A | N/A | N/A | $4444_h$ | $4444_h$ | $4444_h$ |
| $60_h$ | Слово 5 UII | N/A | N/A | N/A | N/A | N/A | $5555_h$ | $5555_h$ |
| $70_h$ | Слово 6 UII | N/A | N/A | N/A | N/A | N/A | N/A. | $6666_h$ |

### I.3 Пример кодека CRC-16

Пример типовой принципиальной схемы кодека CRC-16c показан на рисунке I.2, с использованием полинома и предварительной установки, определенной на рисунке 40 (полином, используемый для расчета CRC-16c, $x^{16} + x^{12} + x^5 + 1$, соответствует Международному стандарту CRC-CCITT, Рекомендация ITU X.25). Приведенная схема применима для расчета как CRC-16, так и CRC-16c.

Для кодирования CRC-16c предварительно заносят в CRC регистр (т. е., Q[15:0]) значение $FFFF_h$, затем подают кодируемые биты данных на вход ДАННЫЕ вперед СЗР по тактам CLK. После ввода всех битов данных регистр Q[15:0] содержит значение CRC-16c в дополнительном коде.

Для декодирования CRC-16c предварительно заносят в регистр CRC (Q[15:0]) значение $FFFF_h$, затем подают принятые данные и биты CRC-16c {данные, CRC-16c} на вход ДАННЫЕ вперед СЗР по тактам CLK. Проверка CRC-16c пройдена, если значение Q[15:0] = $D0F_h$.

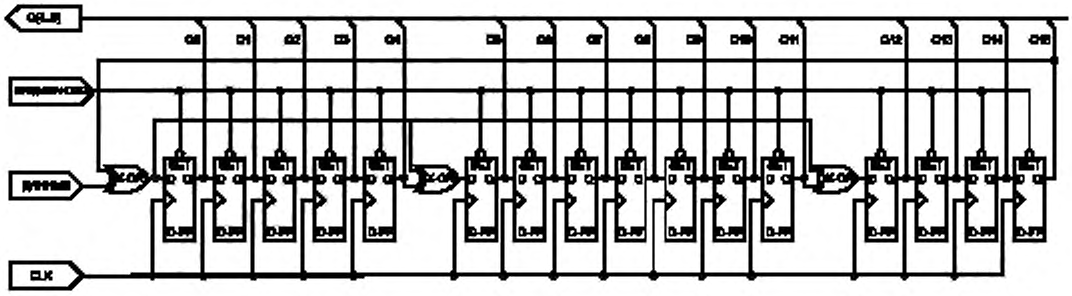



Рисунок I.2 — Пример схемы кодека CRC-16c

**Приложение J**
**(справочное)**

## МОДА 3: метод ASK: Модуляция сигнала в канале связи «УСО-радиочастотная метка»

### J.1 Полоса частот радиосигнала, радиочастотная модуляция и детектирование радиосигнала

Рисунок J.1 демонстрирует полосу частот и модуляцию сигнала для соединения R=>T, генерируемого УСО, а также соответствующую продетектированную радиочастотной меткой огибающую сигнала, для модуляции типа DSB-ASK.

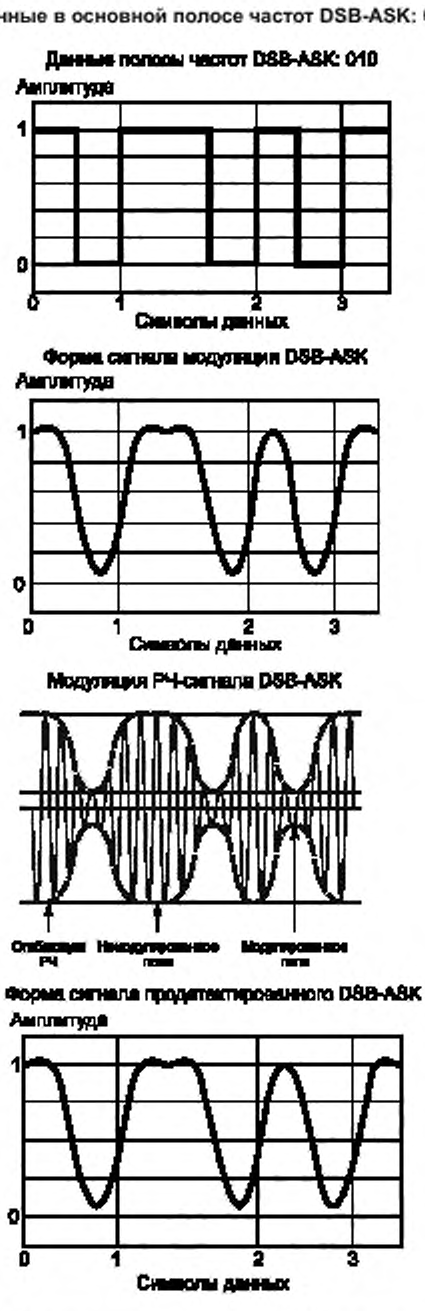



Рисунок J.1 — Модуляция в канале связи «УСО-радиочастотная метка»

Примечание — Модуляция РЧ сигнала DSB-ASK выполняется с соблюдением коэффициента модуляции, указанной в таблице 32.

**Приложение К**
**(справочное)**

## МОДА 3: пример обмена потоками данных

### К.1 Обзор обмена потоками данных

В следующем примере описан обмен данными между УСО и радиочастотной меткой, во время которого УСО читает «пароль уничтожения», хранимый в Резервной памяти радиочастотной метки. В этом примере предполагается, что:

- Радиочастотная метка идентифицирована и находится в состоянии **acknowledged**.
- Резервная память радиочастотной метки блокирована, но не «постоянной блокировкой», это значит, что УСО перед выполнением операции считывания должно передать «пароль доступа» и перевести радиочастотную метку в состояние **secured.**
- Случайные числа, генерируемые радиочастотной меткой (перечислены в не случайной последовательности, из соображений наглядности) имеют вид:
  - CRC16 $XXXX_h$ (радиочастотная метка посылает код StoredCRC перед переходом в состояние **acknowledged**);
  - RN16_1 $1601_h$ (станет маркером подлинности для всей последовательности доступа);
  - RN16_2 $1602_h$;
  - RN16_3 $1603_h$.
- UII радиочастотной метки длиной 64 бита.
- «Пароль доступа» радиочастотной метки $ACCEC0DE_h$;.
- «Пароль уничтожения» радиочастотной метки $DEADC0DE_h$;.
- 1-я половина пароля доступа после операции «ИСКЛЮЧАЮЩЕЕ ИЛИ» с RN16_2 = $ACCE_h \otimes 1602_h = BACC_h$;
- 2-я половина пароля доступа после операции «ИСКЛЮЧАЮЩЕЕ ИЛИ» с RN16_3 = $C0DE_h \otimes 1603_h = D6DD_h$.

### К.2 Содержимое памяти радиочастотной метки и значения поля блокировки

Таблица К.1 демонстрируют пример содержимого памяти радиочастотной метки и значений полей блокирования соответственно.

Таблица К.1 — Содержимое памяти радиочастотной метки

| Банк памяти | Содержимое памяти | Адреса памяти | Значения, хранимые в памяти |
|---|---|---|---|
| TID | TID[15:0] | От $10_h$ до $1F_h$ | $54E2_h$ |
| | TID[31:16] | От $00_h$ до $0F_h$ | $A986_h$ |
| UII | UII[15:0] | От $50_h$ до $5F_h$ | $3210_h$ |
| | UII[31:16] | От $40_h$ до $4F_h$ | $7654_h$ |
| | UII[47:32] | От $30_h$ до $3F_h$ | $BA98_h$ |
| | UII[63:48] | От $20_h$ до $2F_h$ | $FEDC_h$ |
| | Данные StoredPC[15:0] | От $10_h$ до $1F_h$ | $2000_h$ |
| | Коды StoredCRC-16 [15:0] | От $00_h$ до $0F_h$ | $XXXX_h$ как рассчитано (см. приложение I) |
| Резервный | Пароль доступа [15:0] | От $30_h$ до $3F_h$ | $C0DE_h$ |
| | Пароль доступа [31:16] | От $20_h$ до $2F_h$ | $ACCE_h$ |
| | Пароль уничтожения [15:0] | От $10_h$ до $1F_h$ | $C0DE_h$ |
| | Пароль уничтожения [31:16] | От $00_h$ до $0F_h$ | $DEAD_h$ |

Таблица К.2 — Значения поля блокировки

| Пароль уничтожения | | Пароль доступа | | Память UII | | Память TID | | Память пользователя | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | N/A | N/A |

### К.3 Обмен потоками данных и последовательность команд

Обмен потоками данных выполняется в соответствии с процедурой *Access*, показанной на рисунке 49, с добавленной в конце командой *Read*. Последовательность команд УСО и ответов радиочастотной метки имеет вид:

Шаг 1: *Req_RN* [CRC16, CRC-16c]
Радиочастотная метка передает RN 16_1, который становится маркером подлинности для всей последовательности доступа.

Шаг 2: *Req_RN* [CRC-16c маркера подлинности].
Радиочастотная метка передает RN 16_2.

Шаг 3: *Access* [«пароль доступа» [31:16] объединен операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» со значениями RN16_2, CRC-16c маркера подлинности ].
Радиочастотная метка передает маркер подлинности.

Шаг 4: *Req_RN* [CRC-16c маркера подлинности]|.
Радиочастотная метка передает RN16_3.

Шаг 5: *Access* [пароль доступа [15:0] объединен операцией «ИСКЛЮЧАЮЩЕЕ ИЛИ» со значениями RN16_3, CRC-16c маркера подлинности].
Радиочастотная метка передает маркер подлинности.

Шаг 6: *Read* [MemBank =Резервный, WordPtr=$00_h$, WordCount=2, CRC-16c маркера подлинности].
Радиочастотная метка передает «пароль уничтожения».

В таблице К.3 подробно показаны команды УСО и ответы радиочастотной метки. Для ясности CRC-16c был опущен из всех команд и ответов.

Таблица К.3 — Команды УСО и ответы радиочастотной метки

| Шаг | Поток данных | Команда | Параметр и/или данные | Состояние радиочастотной метки |
|---|---|---|---|---|
| 1a: команда *Req_RN* | R => T | 11000001 | 0001 0110 0000 0000 (CRC16=XXXX$_h$) | **acknowledged →open** |
| 1b: ответ радиочастотной метки | T => R | | 0001 0110 0000 0001 (маркер подлинности =$1601_h$) | |
| 2a: команда *Req_RN* | R => T | 11000001 | 0001 0110 0000 0001 (маркер подлинности =$1601_h$) | **open →open** |
| 2b: ответ радиочастотной метки | T => R | | 0001 0110 0000 0010 (RN16_2=$1602_h$) | |
| 3a: команда *Access* | R => T | 11000110 | 1011 1010 1100 1100 (BACC$_h$)<br>0001 0110 0000 0001 (маркер подлинности =$1601_h$) | **open →open** |
| 3b: ответ радиочастотной метки | T => R | | 0001 0110 0000 0001 (маркер подлинности =$1601_h$) | |
| 4a: команда *Req_RN* | R => T | 11000001 | 0001 0110 0000 0001 (маркер подлинности =$1601_h$) | **open →open** |
| 4b: ответ радиочастотной метки | T => R | | 0001 0110 0000 0011 (RN16_2=$1603_h$) | |
| 5a: команда *Access* | R => T | 11000110 | 1101 0110 1101 1101 (D6DD$_h$)<br>0001 0110 0000 0001 (маркер подлинности =$1601_h$) | **open →secured** |
| 5b: ответ радиочастотной метки | T => R | | 0001 0110 0000 0001 (маркер подлинности =$1601_h$) | |

*Окончание таблицы К.3*

| Шаг | Поток данных | Команда | Параметр и/или данные | Состояние радиочастотной метки |
|---|---|---|---|---|
| 6a: команда *Read* | R => T | 11000010 | 00 (MemBank =Резервный)<br>00000000 (WordPtr=пароль уничтожения)<br>00000010 (WordCount =2)<br>0001 0110 0000 0001 (маркер подлинности =$1601_h$) | **secured →secured** |
| 6b: ответ радиочастотной метки | T => R | | 0 (заголовок)<br>1101 1110 1010 1101 ($DEAD_h$)<br>1100 0000 1101 1110 ($C0DE_h$) | |

Примечание — Приведенный выше пример демонстрирует дополнительное использование защиты кодированием пароля доступа. Ответ радиочастотной метки на одном шаге УСО может использовать как параметр в команде для следующего шага.

**Приложение L**
**(справочное)**

## МОДА 3: функции радиочастотной метки

Настоящее приложение приводит краткую информацию о функциях, доступных для радиочастотных меток, которые соответствуют МОДА 3.

### L.1 Опциональные пароли радиочастотной метки

**Пароль уничтожения:** Метка может опционально поддерживать пароль уничтожения. Радиочастотная метка, которая не поддерживает «пароля уничтожения» работает так, как если бы она имела нулевой «пароль уничтожения», который «постоянно блокирован» для считывания/записи. См. 6.3.3.4.1.1.1.

**Пароль доступа:** Метка может опционально поддерживать пароль доступа. Радиочастотная метка, которая не поддерживает «пароль доступа» работает так, как если бы она имела нулевой «пароль доступа», который «постоянно блокирован» для считывания/записи. См. 6.3.3.4.1.1.2.

### L.2 Опциональные банки памяти радиочастотной метки и их размер

**Резервная память:** Резервная память является опциональной. Если радиочастотная метка не поддерживает «пароль уничтожения» и «пароль доступа», то ей не нужно реализовывать Резервную память. Поскольку радиочастотная метка, не поддерживающая пароли, работает как будто она имеет нулевой пароль(и), «постоянно блокированный» для считывания/записи, эти пароли должны быть логически доступны в Резервной памяти, в области, указанной в разделах 6.3.3.4.1.1.10 и 6.3.3.4.1.1.2.

**Память UII:** Память UII необходима, но ее размер опционален. Минимальный размер 32 бита для хранения 16-битового CRC-16 и 16-битового слова PC. Память UII может быть больше 32 битов для хранения UII, опциональная длина которых может быть от 16 до 464 битов с шагом 16 битов, также как обязательного слова *XPC_W1* и опционального слова *XPC_W2*. См. 6.3.3.4.1.1.2.

**Память TID:** Память TID необходима, но ее размер опционален. Минимальный размер памяти TID содержит 8 битов для размещения идентификатора класса по *ГОСТ Р ИСО/МЭК 15963*, также как достаточную для УСО идентифицирующую информацию для однозначной идентификации Команд пользователя и/или опциональных функций, поддерживаемых радиочастотной меткой. Память TID опционально может содержать определенные изготовителем данные. См. 6.3.3.4.1.4.

**Память пользователя:** Память пользователя опциональна. См. 6.3.3.4.1.4, 6.3.3.4.1.4.1 и 6.3.3.4.1.4.2.

### L.3 Опциональные команды радиочастотной метки

**Команды изготовителя:** Радиочастотная метка может поддерживать специальные «команды изготовителя». См. 6.0.9.

**Команды пользователя:** Радиочастотная метка может поддерживать «команды пользователя». См. 6.0.8.

***Access*:** Радиочастотная метка может поддерживать команду *Access*. См. 6.3.3.4.11.3.6.

***BlockWrite*:** Радиочастотная метка может поддерживать команду *BlockWrite*. См. 6.3.3.4.11.3.7.

***BlockErase*:** Радиочастотная метка может поддерживать команду *BlockErase*. См. 6.3.3.4.11.3.8.

***BlockPermalock*:** Радиочастотная метка может поддерживать команду *BlockPermalock*. См. 6.3.3.4.11.3.9.

***Kill*:** Радиочастотная метка может поддерживать команду *Kill*. Обязательна только поддержка третьего МЗР параметра Recom. См. 6.3.3.4.11.3.4.

### L.4 Опциональный формат кодов ошибок радиочастотной метки

Радиочастотная метка может поддерживать специфицированные коды ошибок или неспецифицированные коды ошибок. См. приложение E.

### L.5 Опциональная функциональность радиочастотной метки

Радиочастотная метка может поддерживать UMI* одним из двух методов. См. 6.3.3.4.1.2.2.

Радиочастотная метка может поддерживать второе слово XPC (XPC_W2). См. 6.3.3.4.1.2.2 и 6.3.3.4.1.2.5.

Радиочастотная метка может поддерживать работу с МЗР и вторым МЗР параметра Recom. См. 6.3.3.4.4.7, 6.3.3.4.10 и 6.3.3.4.11.3.4.

Радиочастотная метка может выполнять защиту кодированием. См. 6.3.3.4.11.3.3.

### L.6 Дополнительные возможности радиочастотной метки

Радиочастотная метка может дополнительно поддерживать работу Методом PJM (см. раздел 6).

---

* *Индикатор пользовательской памяти.*

**Приложение М**
**(справочное)**

## Циклический избыточный код (CRC) (16 бит)

### М.1 Метод обнаружения ошибок CRC

МОДА 1: Циклический избыточный код (CRC) рассчитывается по всем данным, содержащимся в сообщении, начиная от флагов до конца данных включительно. Этот CRC используется со стороны УСО для радиочастотной метки, и со стороны радиочастотной метки для УСО, по обстоятельствам.

МОДА 2: Циклический избыточный код (CRC) вычисляется от окончания поля стартового флага. Этот CRC используется только со стороны УСО для радиочастотной метки.

Таблица М.1 — Описание CRC

| Тип CRC | Длина | Полином | Направление | Предустановка | Вычет |
|---|---|---|---|---|---|
| *[10]* | 16 бит | $x^{16} + x^{12} + x^5 + 1 = 8408_h$ | Обратный | $FFFF_h$ | $F0B8_h$ |

Чтобы добавить дополнительную защиту от сдвиговых ошибок, делается дальнейшее преобразование вычисленного CRC. К сообщению для передачи присоединяется дополнительный код вычисленного CRC. Это преобразование включено в нижеприведенный пример.

Для простоты использования 2 байта CRC часто также включают в пересчет при проверке принятых сообщений. В этом случае ожидаемое вычисленное значение для сгенерированного CRC — это остаток от $F0B8_h$.

### М.2 Пример вычисления CRC

Этот пример на языке С иллюстрирует один из методов расчета CRC по заданному набору байтов, включенных в сообщение.

```
#include <stdio.h>"
#define POLYNOMIAL 0x8408 // x^16 + x^12 + x^5 + 1
#define PRESET_VALUE 0xFFFF
#define CHECK_VALUE 0xF0B8
#define NUMBER_OF_BYTES 4 // Пример: 4 бита данных
#define CALC_CRC 1
#define CHECK_CRC 0
void main()
{
     unsigned int current_crc_value;
     unsigned char array_of_databytes[NUMBER_OF_BYTES - 2] = {1, 2, 3, 4,
0x91, 0x39};
     int number_of_databytes = NUMBER_OF_BYTES;
     int calculate_or_check_crc;
     int i, j;
     calculate_or_check_crc = CALC_CRC;
     // calculate_or_check_crc = CHECK_CRC; // Еще один пример
     if (calculate_or_check_crc == CALC_CRC)
     {
         number_of_databytes = NUMBER_OF_BYTES;
     }
     else // проверить CRC
     {
         number_of_databytes = NUMBER_OF_BYTES + 2;
     }
     current_crc_value = PRESET_VALUE;
     for (i = 0; i < number_of_databytes; i++)
     {
```

```
        current_crc_value = current_crc_value ^ ((unsigned
    int)array_of_databytes[i]);
        for (j = 0; j < 8; j++)
        {
            if (current_crc_value & 0x0001)
            {
              current_crc_value = (current_crc_value >> 1) ^ POLYNOMIAL;
            }
            else
            {
              current_crc_value = (current_crc_value >> 1);
            }
        }
    }
    if (calculate_or_check_crc == CALC_CRC)
    {
        current_crc_value = ~current_crc_value;
    }
    else // проверить CRC
    {
        if (current_crc_value == CHECK_VALUE)
        {
            printf ("Checked CRC is ok (0x%04X)\n", current_crc_value);
    }
    else
    {
            printf ("Checked CRC is NOT ok (0x%04X)\n",
      current_crc_value);
    }
  }
}
```

**Приложение N**
**(справочное)**

## Циклический избыточный код (CRC) для МОДА 2 (32 бита)

### N.1 Метод обнаружения ошибки CRC 32

Циклический избыточный код (CRC 32) рассчитывается по всем данным, содержащимся в ответе, от конца флага до конца данных включительно. CRC используется для ответа радиочастотной метки УСО. См. таблицу N.1.

Таблица N.1 — Описание CRC 32

| Длина | Полином | Направление | Предустановка | Вычет |
|---|---|---|---|---|
| 32 бита | $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1 = EDB88320_h$ | Обратный | $FFFFFFFF_h$ | $2144DF1C_h$ |

Чтобы добавить дополнительную защиту от сдвиговых ошибок, делается дальнейшее преобразование рассчитанного CRC. К сообщению для передачи присоединяется дополнительный код рассчитанного CRC. Это преобразование включено в нижеприведенный пример.

Для простоты использования 2 байта CRC часто также включают в пересчет при проверке принятых сообщений. В этом случае ожидаемое расчетное значение для сгенерированного CRC — это остаток от $2144DF1C_h$.

### N.2 Пример расчета CRC 32

Этот пример на языке С иллюстрирует один из методов расчета CRC по заданному набору байтов, включенных в сообщение.

```
#define POLYNOMIAL    0xEDB88320
#define PRESET_VALUE  0xFFFFFFFF
#define CHECK_VALUE   0x2144DF1C
#include <stdio.h>
#include <stddef.h>

unsigned long calculate_CRC32(unsigned long current_crc_value, unsigned
int new_word)
{
 int i;

 current_crc_value = current_crc_value ^ ((unsigned int)new_word);

 for (i = 0; i < 16; i++)
 {
 if (current_crc_value & 0x0001)
 {
 current_crc_value = (current_crc_value >> 1) ^ POLYNOMIAL;
 }
 else
 {
 current_crc_value = (current_crc_value >> 1);
 }
 }
 return(current_crc_value);
}

void main()
{
 unsigned long crc32;
 unsigned long crc_send;

 crc32 = PRESET_VALUE; // установитьCRCв 0xFFFFFFFF
 crc32 = calculate_CRC32(crc32, 0x1234); // типичный ответ радиочастотной метки
```

```
crc32 = calculate_CRC32(crc32, 0x0002);
crc32 = calculate_CRC32(crc32, 0x0003);
crc32 = calculate_CRC32(crc32, 0x0010);
crc32 = calculate_CRC32(crc32, 0x0011);

printf("CRC Results:\n");
printf("-------------\n");

crc_send = ~ crc32; // установлено обращение рассчетного CRC
printf("Send = 0x:08X\n",crc_send);
// если в расчет включается CRC, переданный устройством считывания,
// должен быть результат CHECK_VALUE

crc32 = calculate_CRC32(crc32, (unsigned int)(crc_send) & 0x0000FFFF);
crc32 = calculate_CRC32(crc32, (unsigned int)((crc_send >> 16) & 0x0000FFFF));

crc32 = ~ crc32; // инверсия выходного сигнала для восстановления остатка
printf("Residue = 0x:08X\n",crc32);

if (crc32 == CHECK_VALUE) // должно быть равно CHECK_VALUE
printf("CRC OK \n");
else printf("CRC BAD \n");
}
```

Эта программа после запуска должна выдать следующее:

```
Результаты CRC:
-------------
Отправить = 0xC7D5219F
Зычет = 0x2144DF1C
CRC OK
```

### N.3 Практический пример расчета CRC 32

Этот пример относится к команде с типом ответа «короткий ответ» на считывание двух слов из памяти радиочастотной метки, начиная с адреса $10_h$.

«Специальный идентификатор» радиочастотной метки в этом примере имеет значение $00030002_h$, а номер команды определен как $1234_h$. Ячейка памяти радиочастотной метки может содержать любое значение, в этом примере оно принимается таким, как в таблице N.2:

Таблица N.2 — Пример значений, находящихся в памяти радиочастотной метки

| Адрес | Содержимое |
|---|---|
| $10_h$ | $0010_h$ |
| $11_h$ | $0011_h$ |

Ответ в этом примере состоит из следующих полей:

- «Временная метка», имеющая значение номера переданной команды '$1234_h$';
- «Специальный идентификатор» радиочастотной метки '$00030002_h$', передаваемый двумя словами, первым передается младшее слово;
- Содержимое двух прочитанных ячеек памяти, первым передается младший адрес: '$0010_h$' '$0011_h$';
- CRC: '$C7D5219F_h$' где '$219F_h$' — младшее слово, которое передается первым.

Ответ передается в следующем виде: 1234 0002 0003 0010 0011 219F C7D5

Таблица N.3 — Пример расчета CRC в УСО

| Шаг | Входное значение | Рассчитанное в УСО значение CRC |
|---|---|---|
| 1 | Инициализировано | $FFFFFFFF_h$ |
| 2 | $1234_h$ | $094A9040_h$ |
| 3 | $0002_h$ | $7399A576_h$ |

*Окончание таблицы N.3*

| Шаг | Входное значение | Рассчитанное в УСО значение CRC |
|---|---|---|
| 4 | $0003_h$ | $B52DBBB2_h$ |
| 5 | $0010_h$ | $815B13BD_h$ |
| 6 | $0011_h$ | $C7D5219F_h$ |
| 7 | $219F_h$ | $41D9D52A_h$ |
| 8 | $C7D5_h$ | $2144DF1C_h$ |

Примечания

1 CRC, которое в конечном счете передается, 'C7D5219Fh' присутствует в виде 'Рассчитанное CRC в устройстве опроса' на шаге 6, после каждого из слов, данных в ответе и включенных в расчет.

Последующий расчет со следующими двумя словами (отправленный CRC) приводит к окончательному результату, значением которого должен быть остаток '$2144DF1C_h$'.

2 CRC передается как два слова, младшее слово передается первым.

3 Каждое слово передается начиная с младшего значащего бита.

**Приложение О**
**(справочное)**

## Известные возможные перекрестные взаимодействия между МОДА, определенными в настоящем стандарте

Перекрестные взаимодействия неизвестны.

Примечание — Перекрестные взаимодействия маловероятны, так как эти три режима совершенно различны и у всех обеспечивается высокий уровень целостности данных (ITF, кадрирование, CRC).

**Приложение ДА**
**(справочное)**

## Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте

Таблица ДА.1

| Обозначение ссылочного национального стандарта | Степень соответствия | Обозначение и наименование ссылочного международного стандарта |
|---|---|---|
| ГОСТ Р ИСО/МЭК 7816-6—2013 | IDT | ISO/IEC 7816-6:2004* «Карты идентификационные. Карты на интегральных схемах. Часть 6. Межотраслевые элементы данных для обмена» |
| ГОСТ Р ИСО/МЭК 15693-1—2013 | IDT | ISO/IEC 15693-1:2010 «Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 1. Физические характеристики» |
| ГОСТ Р ИСО/МЭК 15693-2—2013 | IDT | ISO/IEC 15693-2:2006 «Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 2. Воздушный интерфейс и инициализация» |
| ГОСТ Р ИСО/МЭК 15693-3—2011 | IDT | ISO/IEC 15693-3:2009 «Карты идентификационные. Карты на интегральных схемах бесконтактные. Карты удаленного действия. Часть 3. Антиколлизия и протокол передачи данных» |
| ГОСТ Р ИСО/МЭК 15963—2011 | IDT | ISO/IEC 15963:2009 «Информационные технологии. Радиочастотная идентификация для управления предметами. Уникальная идентификация радиочастотных меток» |
| ГОСТ Р ИСО/МЭК 18046—2009 | IDT | ISO/IEC 18046:2006** «Автоматическая идентификация. Идентификация радиочастотная. Методы испытаний технических характеристик устройств радиочастотной идентификации» |
| ГОСТ Р ИСО/МЭК 19762-1—2011 | IDT | ISO/IEC 19762-1:2008 «Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 1. Общие термины в области АИСД» |
| ГОСТ Р ИСО/МЭК 19762-2—2011 | IDT | ISO/IEC 19762-2:2008 «Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 2. Оптические носители данных (ОНД)» |
| ГОСТ Р ИСО/МЭК 19762-3—2011 | IDT | ISO/IEC 19762-3:2008 «Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 3. Радиочастотная идентификация» |
| ГОСТ Р ИСО/МЭК 19762-4—2011 | IDT | ISO/IEC 19762-4:2008 «Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 4. Общие понятия в области радиосвязи» |
| ГОСТ Р 56914—2016 | IDT | ISO/IEC TR 18047-3:2011 «Информационные технологии. Методы испытаний на соответствие устройств радиочастотной идентификации. Часть 3. Методы испытаний радиоинтерфейса для связи на частоте 13,56 МГц» |

Примечание — В настоящей таблице использовано следующее условное обозначение степени соответствия стандартов:

- IDT — идентичные стандарты.

* *Действует ISO/IEC 7816-6:2016.*

** *Действуют ISO/IEC 18046-1:2011, ISO/IEC 18046-2:2011, ISO/IEC 18046-3:2012, ISO/IEC 18046-4:2015.*

**Приложение ДБ**
**(справочное)**

## Сведения о соответствии международных и русских команд, параметров и технических терминов

Сведения о соответствии международных и русских команд приведены в таблице ДБ.1.

Таблица ДБ.1

| Наименование команды на английском языке | Наименование команды на русском языке |
|---|---|
| *Select* | *Выбор* |
| *Read* | *Считывание* |
| *Write* | *Запись* |
| *BlockWrite* | *Запись блока* |
| *BeginRound* | *Начать цикл опроса* |
| *BlockErase* | *Стереть блок* |
| *Kill* | *Уничтожение* |
| *Access* | *Доступ* |
| *Lock* | *Блокировка* |
| *BlockPermalock* | *Постоянная блокировка блока* |
| *NextSlot* | *Следующий слот* |
| *ResizeRound* | *Калибровка цикла* |

Сведения о соответствии международных и русских параметров приведены в таблице ДБ.2.

Таблица ДБ.2

| Наименование параметра на английском языке | Наименование параметра на русском языке |
|---|---|
| action | Действие |
| blockrange | Диапазон блока |
| channel/mute ratio | Коэффициент загруженности канала |
| command length | Длина кода команды |
| EOF | Завершающий кадр |
| handle, descriptor | Маркер подлинности, дескриптор |
| implement | Включать в себя средства реализации |
| invalid | Недопустимый |
| invalid command | Недопустимая команда |
| item | Предмет учета |
| Length | Длина |
| Mask | Маска |
| MemBank | Банк памяти |

*Окончание таблицы ДБ.2*

| Наименование параметра на английском языке | Наименование параметра на русском языке |
|---|---|
| mute ratio | Коэффициент блокировок |
| PacketCRC | Код PacketCRC |
| PacketPC | Данные PacketPC |
| pilot signal | Пилотный сигнал (предварительный тоновый сигнал) |
| pivot interval | Опорный интервал |
| Plaintext | Открытый текст |
| Pointer | Указатель |
| Pointerlength | Длина указателя |
| Read/Lock | Считывание/блокировка |
| SOF | Стартовый кадр |
| StoredCRC | Код StoredCRC |
| StoredPC | Данные StoredPC |
| sub-block | Минимальная область адресуемой памяти |
| target | Цель |
| temporary (fully) mute state | Состояние «режим ожидания» (радиочастотной метки) |
| temporary mute state | Состояние «Временно заблокирована» |
| to loadmodulate | Передавать (по отношению к радиочастотной метке) |
| truncate | Усечение |
| User memory bank | Банк пользовательской памяти |
| WordCount | Счетчик слов |
| WordPtr | Слово указателя |
| WordPtrlength | Слово длины указателя |

Сведения о соответствии международных и русских технических терминов приведены в таблице ДБ.3.

Таблица ДБ.3

| Наименование технического термина на английском языке | Наименование технического термина на русском языке |
|---|---|
| cover-coding | Защита кодированием |
| direct-sequence [DSSS] systems | Системы широкополосной модуляции с прямым расширением спектра |
| encoder/decoder | Кодек |
| frequency-hopping [FHSS] systems | Система, использующая алгоритм псевдослучайной перестройки рабочей частоты [FHSS]) |
| full-duplex communications | Дуплексная связь |
| half-duplex communications | Полудуплексная связь |
| implement | Включать в себя средства реализации |
| invalid | Недопустимое значение |

*Окончание таблицы ДБ.3*

| Наименование технического термина на английском языке | Наименование технического термина на русском языке |
|---|---|
| item | Предмет учета |
| kill a tag | «Уничтожение» метки, перевод в состояние «Уничтожена» |
| link | Канал связи |
| locked memory | Заблокированная память |
| LSB и MSB | Младший (старший) значащий бит (МЗБ и СЗБ соответственно) |
| MODE | МОДА |
| Modified Frequency Modulation | Модифицированная частотная модуляция |
| permalocked memory | «Постоянно заблокированная» память |
| phase jitter modulation (PJM) | Фазовая модуляция колебаний (PJM) |
| physical layer | Физический уровень |
| pilot signal | Пилотный сигнал (предварительный тоновый сигнал) |
| pivot interval | Опорный интервал |
| plaintext | Открытый текст |
| pread-spectrum systems | Широкополосная система |
| read/write command | Команда считывания/записи |
| recommissioning | «Ввод в действие» |
| specific command | Целевая команда (для работы с отдельными метками) |
| sub-block | Минимальная область адресуемой памяти |
| tag | Радиочастотная метка, РЧ метка |
| temporary (fully) mute state | Состояние «Режим ожидания» (радиочастотной метки) |
| temporary mute state | Состояние «Временно заблокирована» |
| to loadmodulate | Передавать (по отношению к радиочастотной метке) |
| valid command | Действительная команда |

**Приложение ДВ**
**(справочное)**

## Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта

Таблица ДВ.1

| Структура настоящего стандарта | Структура международного стандарта ИСО/МЭК 18000-3:2010 |
|---|---|
| Приложение ДА Сведения о соответствии ссылочных национальных стандартов международным стандартам, использованным в качестве ссылочных в примененном международном стандарте | — |
| Приложение ДБ Сведения о соответствии международных и русских команд, параметров и технических терминов | — |
| Приложение ДВ Сопоставление структуры настоящего стандарта со структурой примененного в нем международного стандарта | — |
| Примечание — Сопоставление структуры стандартов приведено, начиная с приложения ДА, так как предыдущие разделы стандартов идентичны. | |

## Библиография

[1] *ИСО/МЭК 18000-1. Информационные технологии. Радиочастотная идентификация для управления предметами. Часть 1. Рекомендуемая архитектура и определение параметров, подлежащих стандартизации*

[2] *Стандарты данных радиочастотной метки EPC (версия 1.3 и выше) (EPCglobal Tag Data Standards (Version 1.3 and above) EPCglobal™)*

[3] *ИСО/МЭК 15962, Информационные технологии. Радиочастотная идентификация (РЧИ) для управления предметами. Протокол данных: правила кодирования данных и логические функции памяти*

[4] *ГОСТ Р ИСО/МЭК 7810—2015, Карты идентификационные. Физические характеристики*

[5] *ИСО/МЭК 15961-3:2019, Информационные технологии. Протокол передачи данных для радиочастотной идентификации для управления предметами. Часть 3. Структуры данных радиочастотной идентификации*

[6] *ИСО/МЭК 15961-2, Информационные технологии. Протокол передачи данных для радиочастотной идентификации для управления предметами. Часть 2. Регистрация структур данных радиочастотной идентификации*

[7] *ИСО/МЭК 13239. Информационные технологии. Телекоммуникации и обмен информацией между системами. Процедуры высокого уровня управления каналом передачи данных (HDLC)*

[8] *ИСО/МЭК 15961, Информационные технологии. Радиочастотная идентификация (РЧИ) для управления предметами. Протокол передачи данных: интерфейс приложения*

УДК 681.5.015:621.3:006.354 ОКС 35.040

Ключевые слова: информационные технологии, радиочастотная идентификация, радиоинтерфейс, радиочастотная метка, устройство опроса

БЗ 12—2019/17

Редактор *В.Н. Шмельков*
Технический редактор *И.Е. Черепкова*
Корректор *М.В. Бучная*
Компьютерная верстка *Л.А. Круговой*

Сдано в набор 20.11.2019 Подписано в печать 30.12.2019. Формат 60×84 1/8. Гарнитура Ариал.
Усл. печ. л. 20,00. Уч.-изд. л. 17,00.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении во ФГУП «СТАНДАРТИНФОРМ» для комплектования Федерального информационного фонда стандартов, 117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru gost.e-ecolog.ru

gost.e-ecolog.ru