INTERNATIONAL STANDARD

# ISO 14223-2

First edition
2010-08-15

# Radiofrequency identification of animals — Advanced transponders —

## Part 2:
## Code and command structure

*Identification des animaux par radiofréquence — Transpondeurs évolués —*

*Partie 2: Code et structure de commande*

Reference number
ISO 14223-2:2010(E)

# Contents

Page

# Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

ISO 14223-2 was prepared by Technical Committee ISO/TC 23, *Tractors and machinery for agriculture and forestry*, Subcommittee SC 19, *Agricultural electronics*.

ISO 14223 consists of the following parts, under the general title *Radiofrequency identification of animals — Advanced transponders*:

— *Part 1: Air Interface*

— *Part 2: Code and command structure*

The following part is under preparation:

— *Part 3: Applications*

# Introduction

This part of 14223 specifies the communication interface of the radio frequency (RF) system for advanced transponders for animals. The technical concept of advanced transponders for animal identification described is based upon the principle of radio frequency identification (RFID) and is an extension of the standards ISO 11784 and ISO 11785. Apart from transmission of the (unique) identification code of animals, the application of advanced technologies facilitates the storage and retrieval of additional information (integrated database), the implementation of authentication methods and the reading of data from integrated sensors, etc.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents concerning the methods of transmission referred to throughout the document.

ISO takes no position concerning the evidence, validity and scope of these patent rights.

The holders of these patent rights have assured ISO that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO. Information may be obtained from:

N.V. Nederlandsche Apparatenfabriek “Nedap”
Parallelweg 2
NL-7141 DC Groenlo
The Netherlands

Texas Instruments Deutschland GmbH
Haggerstrasse 1
D-85356 Freising
Germany

NXP Semiconductors
Mikron-Weg 1
A-8101 Gratkorn
Austria

EM Microelectronic-Marin SA
Sors 3
CH-2074 Marin
Switzerland

Atmel Germany GmbH
P.O. Box 3535
74025 Heilbronn
Germany

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. ISO shall not be held responsible for identifying any or all such patent rights.

# Radiofrequency identification of animals — Advanced transponders —

# Part 2: Code and command structure

## 1 Scope

This part of ISO 14223 specifies the code and command structure between the transceiver and the advanced transponder used in the radiofrequency identification of animals, this specification being fully backwards-compatible with those of ISO 11784 and ISO 11785. As a direct extension of ISO 11785, it is intended to be used in conjunction with that International Standard.

## 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 11784, *Radio frequency identification of animals — Code structure*

ISO 11785, *Radio frequency identification of animals — Technical concept*

ISO/IEC 7816-6, *Identification cards — Integrated circuit cards — Part 6: Interindustry data elements for interchange*

ISO 24631-1, *Radiofrequency identification of animals — Part 1: Evaluation of conformance of RFID transponders with ISO 11784 and ISO 11785 (including granting and use of a manufacturer code)*

## 3 Conformance

### 3.1 Transponder

For conformance with this part of ISO 14223 to be claimed, a transponder shall be FDX-ADV or HDX-ADV.

NOTE Nothing in this International Standard prevents a transponder being of more than one type, although for technical reasons, it is unlikely that such transponders are ever marketed.

### 3.2 Transceiver

For conformance with this part of ISO 14223 to be claimed, a transceiver shall support both FDX-ADV and HDX-ADV. When in the inventory mode, the transceiver shall alternate between FDX-ADV and HDX-ADV interrogation. The transceiver shall move back to ISO 11785 mode after completion of the advanced operation.



## 4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

**4.1**
**advanced transponder**
transponder conforming to ISO 14223, downward compatible according to ISO 11784 and ISO 11785, with facilities for storage and retrieval of additional data, integrated sensors, etc.

**4.2**
**advanced mode**
operating method of the advanced transponder after reception of a valid command

**4.3**
**anticollision sequence**
algorithm used to prepare for and handle a dialogue between transceiver and one or more transponders out of several in its energizing field

**4.4**
**byte**
eight bits of data designated b1 to b8, from the most significant bit (MSB, b8) to the least significant bit (LSB, b1)

## 5 Abbreviated terms

| | |
|---|---|
| BSS | block security status |
| CRC | cyclic redundancy check |
| CRCT | response cyclic redundancy check flag |
| DSFID | data storage format identifier |
| EOF | end of frame |
| FDX | full duplex |
| IC | integrated circuit |
| ICR | integrated circuit reference number |
| HDX | half duplex |
| LSB | least significant bit |
| MFC | integrated circuit manufacturer code |
| MSB | most significant bit |
| MSN | manufacturer serial number |
| NOB | number of blocks per page |
| NOP | number of pages |
| NOS | number of slots |
| NRZ | non-return to zero |

NSS number of sensors

RF radio frequency

RTF reader talk first

RFU reserved for future use

SOF start of frame

UID unique identifier (includes MFC and MSN)

# 6 Transmission protocol

## 6.1 Basic elements

The advanced transmission protocol defines the mechanism for exchanging instructions and data between the transceiver and the transponders, in both directions.

It is based on the following concepts.

— The transponders are by default conformant with ISO 11784 and ISO 11785. This shall be evaluated conformant with ISO 24631-1. For advanced instructions the transceiver has the ability to communicate with a transponder in the advanced mode. In this mode the transponder is communicating in RTF mode and does not start to respond unless it has received and decoded a valid request from the transceiver.

— The transponders are uniquely identified by 48 bit UID, programmed at the manufacture of the integrated circuit. The UID coding is defined in 6.2.

— An identification code of 64 bits according to ISO 11784 is stored in page 0 (the four blocks given in Table 11 can be used to store the full ISO 11785 protocol) of the user memory area (blocks 0 to 3). This identification code shall be programmed and locked by the transponder issuer in order to avoid manipulations.

The advanced mode protocol is based on

— a request from the transceiver to the transponder, and

— a response from the transponder to the transceiver.

The protocol is bit-oriented. The number of bits transmitted after a SOF depends on the respective request and response.

Flags are used for the control of request and response format. The setting of the flags indicates either request and response variants (e.g. number of slots) or the presence of optional fields. In the case of optional fields, when the flag is set to one (1), the field is present. When the flag is reset to zero (0), the field is absent.

RFU flags shall be set to zero (0).

## 6.2 Unique identifier

The UID is used for addressing each transponder uniquely and individually.

The length of the UID is 48 bits, the format of the UID is presented in Table 1. The IC manufacturer is responsible for setting the UID as defined by this part of ISO 14223 and for ensuring the uniqueness of the MSN.

**Table 1 — UID format**

| MSB | LSB |
|---|---|
| 48 .................... 41 | 40 .................... 1 |
| IC manufacturer code<br>(MFC) | IC manufacturer serial number<br>(MSN) |

The UID shall comprise

— the 8 bit MFC, according to ISO/IEC 7816-6, and

— the 40 bit MSN, a unique serial number assigned by the IC manufacturer.

## 6.3 Request format

A request consists of the following elements:

— SOF;

— flags;

— command;

— parameters (depending on the command);

— data (depending on the command);

— CRC (optional);

— EOF.

The general request format is presented in Table 2.

**Table 2 — General request format**

| | | | | | | |
|---|---|---|---|---|---|---|
| SOF | flags | command | parameters | data | CRC | EOF |

Each request starts with a SOF. The fields are transmitted successively from the first field (flags) to the last field (e.g. CRC). All fields are transmitted LSB first. At the end of a request, an EOF is appended.

The allocation of the LSB and MSB for each field of the request format is shown in Table 3.

**Table 3 — Allocation of LSB and MSB to the request fields**

| | LSB MSB | LSB MSB | LSB MSB | LSB MSB | LSB MSB | |
|---|---|---|---|---|---|---|
| SOF | Field 1 (5 bits)<br>(Flags 1 … 5) | Field 2 (6 bits)<br>(Command) | Field 3<br>(Parameters) | Field 4<br>(Data) | Field 5 (16 bits)<br>(CRC) | EOF |

### 6.4 Response format

A response consists of the following elements:

- SOF pattern;
- error flag;
- error code;
- data (depending on the command);
- CRC (is optional depending on command and flag settings);
- EOF pattern.

The format of the general response if there is no error is presented in Table 4 and that of the general response if there is an error in Table 5.

**Table 4 — General response format if no error**

| SOF | Error flag 0 | Data | CRC | EOF |
|---|---|---|---|---|

**Table 5 — General response format if error**

| SOF | Error flag 1 | Error code | CRC | EOF |
|---|---|---|---|---|

Each response begins with a SOF. The subsequent fields are transmitted successively from the first field (Flag) to the last field (e.g. CRC). All fields are transmitted LSB first. At the end of a response, an EOF is appended.

The allocation of the LSB and MSB for each field of the response format is shown in Table 6.

**Table 6 — Allocation of LSB and MSB to response fields**

| | LSB MSB | LSB MSB | LSB MSB | |
|---|---|---|---|---|
| SOF | Field 1 (1 bit)<br>(Error flag) | Field 2 ($\geqslant$3 bits)<br>(Error code or data) | Field 3 (16 bits)<br>(CRC) | EOF |

### 6.5 Request flags

#### 6.5.1 General

In each request, five flags are used, with flag b1 to be transmitted first. The specific meaning of the request flags depends on the context. The meaning of request flags b1 to b3 is explained in Table 7, that of b4 and b5 where the inventory flag is *not* set in Table 8 and that where the inventory flag *is* set in Table 9.

**Table 7 — Meaning of request flags b1 to b3**

| Bit | Flag name | Value | Description |
|---|---|---|---|
| b1 | PEXT (protocol extension) flag | 0 | No protocol format extension |
| | | 1 | RFU |
| b2 | INV (inventory) flag | 0 | The meaning of flags b4 to b5 is according to Table 8 |
| | | 1 | The meaning of flags b4 to b5 is according to Table 9 |
| b3 | CRCT | 0 | CRC shall *not* be appended to the transponder response |
| | | 1 | CRC shall be appended to the transponder response |

**Table 8 — Request flags b4 to b5 definition when inventory flag is *not* set**

| Bit | Flag name | Value | Description |
|---|---|---|---|
| b4 | SEL (select) flag | 0 | Request shall be executed by any transponder according to the setting of the address flag. |
| | | 1 | Request shall be executed only by transponder in selected state. The address flag shall be set to 0 and the UID field shall not be included in the request. |
| b5 | ADR (address) flag | 0 | Request is not addressed. UID field is not included. It shall be executed by any transponder. |
| | | 1 | Request is addressed. UID field is included. It shall be executed only by that transponder whose UID matches the UID specified in the request. The SEL flag shall be set to 0. |

**Table 9 — Request flags b4 to b5 definition when the inventory flag is set**

| Bit | Flag name | Value | Description |
|---|---|---|---|
| b4 | RFU | 0 | — |
| b5 | NOS flag | 0 | 16 slots |
| | | 1 | 1 slot |

A further description of these flags is given in 6.5.2 to 6.5.4.

### 6.5.2 NOS flag

The NOS flag (see Table 10) is used by the INVENTORY command to select the number of slots during execution of the anticollision sequence.

**Table 10 — Meaning of NOS flag**

| NOS flag | Meaning only for INVENTORY command |
|---|---|
| 0 | 16 slots |
| 1 | 1 slot |

### 6.5.3 SEL flag and ADR flag

The SEL flag and ADR flag are used by all commands except the INVENTORY and READ UID commands.

When both the ADR flag and the SEL flag are set to 0, the request shall not contain a UID. Any transponder in the ready state receiving such a request shall execute it (if possible) and shall return a response to the transceiver as specified by the command description.

When the ADR flag is set to 1 (addressed mode), the request shall contain the UID of the addressed transponder. Independent of the state, any transponder receiving such a request shall compare the received UID (address) to its own ID. If it matches, it shall execute it (if possible) and return a response to the transceiver as specified by the command description. If it does not match, it shall remain silent and keep its current state.

When the SEL flag is set to 1 (selected mode), the request shall not contain a transponder UID. Only the transponder in the selected state receiving such a request shall execute it (if possible) and shall return a response to the transceiver as specified by the command description. Other transponders not in the selected state shall keep their current state and be silent. The combination of ADR and SEL flag is not supported.

Table 11 gives an overview of the meaning of the SEL flag and ADR flag.

**Table 11 — Meaning of SEL flag and ADR flag**

| SEL | ADR | Meaning for all commands except INVENTORY and READ UID |
|---|---|---|
| 0 | 0 | No UID is attached. All transponders in the ready state shall execute this command. |
| 0 | 1 | The UID is attached. Only the transponder with corresponding UID shall execute this command. |
| 1 | 0 | No UID is attached. Only the transponder in the selected state shall execute this command. |
| 1 | 1 | RFU |

### 6.5.4 CRCT flag

The CRCT flag specifies whether or not the transponder is to attach a CRC in its response. The CRC implementation on the transponder is mandatory.

## 6.6 Response flag and error code

The error flag indicates whether or not the transponder has detected an error (see Table 12). If it is set to 1, the response error field shall be returned according to Table 13.

If the transponder does not support specific error codes (as listed in Table 13) it shall answer with the error code 7 “unknown error”.

**Table 12 — Error flag**

| Error flag | Meaning |
|---|---|
| 0 | No error |
| 1 | Error detected |

**Table 13 — Error code**

| Code | Description |
|---|---|
| 0 | RFU |
| 1 | RFU |
| 2 | RFU |
| 3 | The specified block is not available (doesn't exist). |
| 4 | The specified block is secured and its content cannot be accessed. |
| 5 | The specified block was not successfully programmed/locked. |
| 6 | RFU |
| 7 | Unknown error. |

The general response format in case of an error response shall be according to the Table 14 format; commands not supporting error responses are excluded. In the case of an unsupported command there will be no response.

**Table 14 — General response format (with error)**

| SOF | Error flag | Error code | CRC | EOF |
|---|---|---|---|---|
| | 1 | | (optional) | |
| | 1 bit | 3 bits | 16 bits | |

## 6.7 Error handling

In the case of any of the following errors, there will be no response:

— unsupported command;

— the inventory flag is set to 1;

— wrong manufacturer code in a customer command;

— CRC error;

— RFU bit ≠ 0;

— command format error.

## 6.8 Block security status (BSS)

The BSS is sent back by the transponder as a parameter in the response to a transceiver request as specified in 10.4 (e.g. READ SINGLE BLOCK WITH SECURITY STATUS). It is coded on four bits, as explained by Table 15.

It is an element of the protocol. There is no implicit or explicit assumption that the four bits are actually implemented in the physical memory structure of the transponder.

**Table 15 — Block security status**

| Bit | Meaning | Value | Description |
|---|---|---|---|
| Bit 1 | Block lock bit | 0 | Not locked |
| | | 1 | Locked |
| Bit 2 to Bit 4 | RFU | 0 | Reserved for future use |

## 6.9 Start of frame pattern (SOF)

### 6.9.1 Transceiver request

The transceiver request always starts with a SOF pattern. The SOF pattern is defined in ISO 14223-1.

### 6.9.2 Transponder response

The transponder response in the advanced mode always starts with a SOF pattern. The SOF pattern is defined in ISO 14223-1.

## 6.10 Cyclic redundancy check (CRC)

The CRC is according to ISO 11785. The initial register content shall be all zeros: “0000”. The CRC length is 16 bits.

The request CRC is calculated on all bits of the request after the SOF up to the CRC field. The transponder shall detect the presence of the request CRC by the number of received bits.

The response CRC is calculated on all bits of the response after the SOF up to the CRC field. Whether the response CRC is appended to a response depends on the setting of the CRCT flag.

Upon reception of a request from the transceiver containing the CRC, the transponder shall verify the CRC value. If it is invalid, it shall discard the frame and remain silent.

Upon reception of a response from the transponder containing CRC, the transceiver shall verify the CRC value. If it is invalid, any actions to be performed are left to the responsibility of the transceiver designer.

An example of CRC calculation is given in of ISO 11785:1996, Annex B.

## 6.11 Data storage format identifier (DSFID)

The DSFID indicates how the data is structured in the transponder memory. The different data structures will be defined in ISO 14223-3.

It may be programmed and locked by the respective commands. It is coded on one byte. It allows for instant knowledge on the logical organization of the data.

If the programming and locking commands are not supported by the transponder, the transponder shall not respond to these commands.

If the DSFID is not supported or has not been programmed, the transponder shall return the default value “00” in answer to the commands requesting its value.

# 7 Memory organization

## 7.1 General

The physical transponder memory is divided into three logical sections.

— The first logical memory section contains the system data (see 10.4.5).

— The second logical memory section contains the ISO 11784 and ISO 11785 code (see Table 16) and user data. These memory blocks are part of the user memory page 0.

— The third logical memory section contains user data (see Table 17). These memory blocks are part of the user memory page ⩾ 1.

The user memory is organized in pages, each page containing up to 256 blocks; the number of blocks per page shall be the same for all pages. Block size is 32 bits, with up to 256 pages, each with up to 256 blocks; a maximum user memory of 256 kByte can be addressed.

## 7.2 User data memory — Page 0

Blocks 0 to 3 of the user memory Page 0 are used to store the ISO 11784 and ISO 11785 code (see Table 16). It shall be locked just after programming to ensure any manipulation of this code after programming during the lifetime of the animal transponder. The first block accessible for users is block 4.

**Table 16 — User data memory organization (page 0)**

| Block Address | Size | Description |
|---|---|---|
| 0 to 3 | 4 times 32 bits | User data<br>(Bits 1 to 128 reserved for ISO 11785 identification data or frame) |
| 4 | 32 bits | User data |
| ... | ... | … |
| NOB | 32 bits | User data |

## 7.3 User data memory — Extended memory (⩾ page 1)

In the case of an advanced transponder having the means to handle sensor data, the blocks in the upper user memory are reserved for sensor data. The sensor hardware information is defined in the highest block (NOB). The meaning of the sensor data is defined in ISO 14223-3. The size of the sensor data area, i, is defined through the DSFID. In that case, the user data memory organization is described in Table 17. This memory can be addressed by extended memory commands.

**Table 17 — User data memory organization (page 1 to NOP)**

| Block Address | Size | Description |
|---|---|---|
| 0 | 32 bits | User data |
| ... | ... | … |
| NOB-i-1 | 32 bits | User data |
| NOB-i | 32 bits | User data [sensor data] |
| … | … | … |
| NOB-1 | 32 bits | User Data [sensor data] |
| NOB | 32 bits | User Data [sensor hardware information] |

# 8 Transponder states

## 8.1 General

A transponder can be in one of the six following states:

— RF-off;

— wait;

— ISO 11785;

— ready;

— selected (optional);

— quiet.

The support of the RF-off, wait, ISO 11785, ready and quiet states is mandatory. The support of the selected state is optional.

After powering up, the FDX-ADV transponder stays in the wait state for a defined time-out period.

For advanced transponders, a change from the wait stage to the ready state takes place on SOF.

When the transponder cannot process a transceiver request (e.g. CRC error), it shall stay in its current state.

## 8.2 RF-off state

The transponder is in the RF-off state when it is not activated by the transceiver.

## 8.3 ISO 11785 state

This is the state where the ISO 11785 protocol is executed.

## 8.4 Wait state

The wait represents a transition phase in which the transponder can be switched to the advanced mode.

## 8.5 Ready state

The transponder moves to the ready state after receiving a valid request; in this way it is switched to advanced mode.

## 8.6 Quiet state

A transponder enters the quiet state after receiving the STAY QUIET command issued to the transponder. In the quiet state, the transponder shall process any request for which the ADR flag is set.

The transponders shall enter the quiet state if it is in the selected state and receives a SELECT command addressed to another transponder.

## 8.7 Selected state

A transponder enters the selected state after receiving the SELECT command with a matching UID. In the selected state, the respective commands with SEL flag = 1 are valid only for the selected transponder.

Only one transponder shall be in the selected state at any time. If a first transponder is in the selected state, and a second transponder will be selected by the SELECT command, the first transponder shall enter automatically the quiet state.

## 8.8 State diagram

In each of the states, except for the wait state, the transponder accepts only dedicated commands. All other commands are ignored. Figure 1 presents the transponder state diagram.

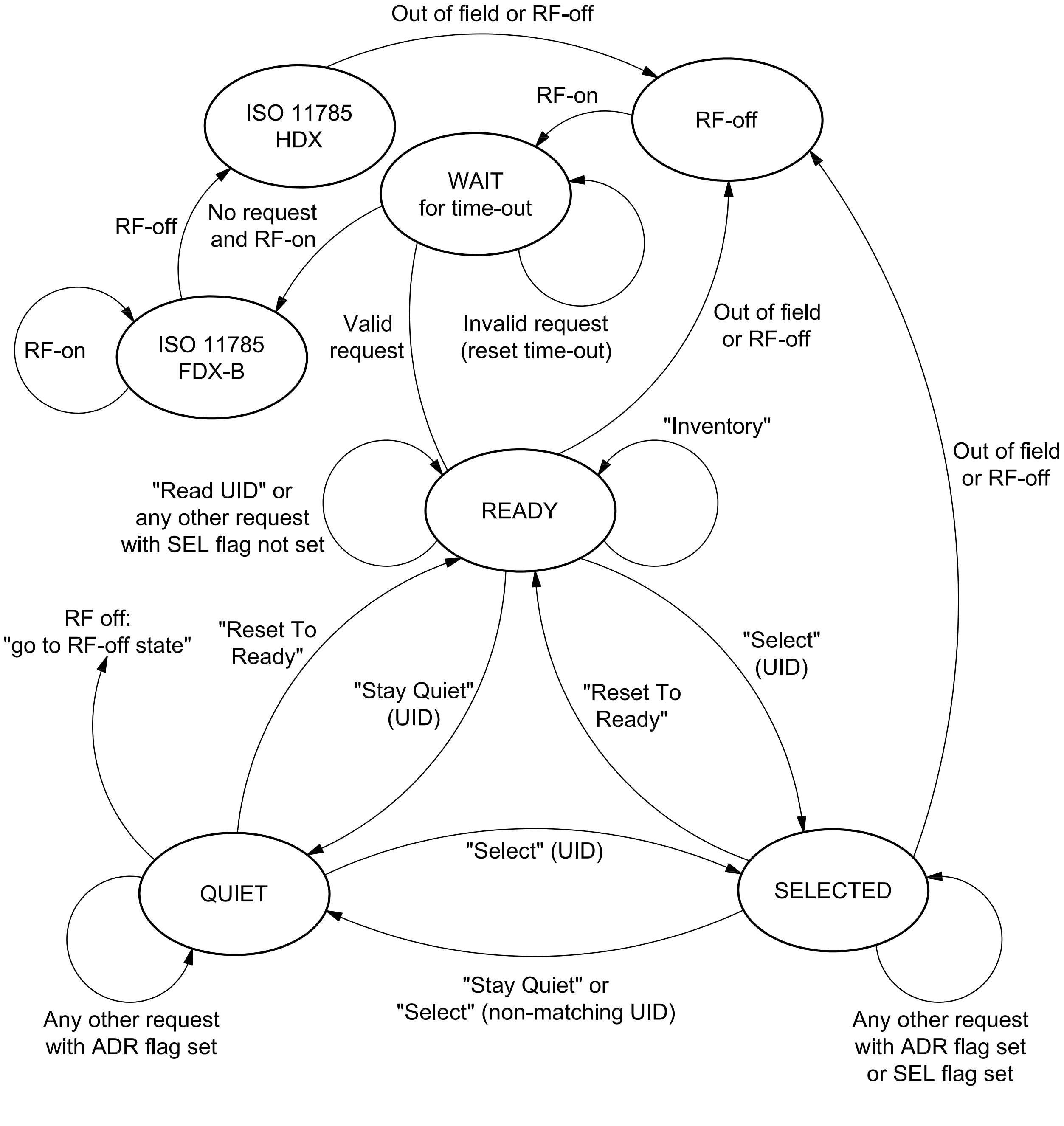



**Figure 1 — Transponder state diagram**

# 9 Anticollision

## 9.1 General

The purpose of the anticollision sequence is to make an inventory of the transponders present in the transceiver field by their UID.

The transceiver is the master of the communication with one or multiple transponders. It starts the anticollision sequence by issuing the INVENTORY request.

The transponder shall send its response in the slot determined or shall not respond, according to the algorithm described in 9.3.

## 9.2 Request parameters

When issuing the INVENTORY command, the transceiver shall set the NOS flag to the desired setting (1 or 16 slots) and add after the command field the mask length and the mask value.

The mask length, $n$, indicates the number of significant bits of the mask value. It can have any value between 0 and 44 when 16 slots are used (see Table 19) and any value between 0 and 48 when 1 slot is used (see Table 18).

**Table 18 — Inventory request format (mask length 1 slot)**

| SOF | Flags | Command | Mask length, $n$<br>1 slot $0 \leqslant n < 48$ | Mask value | CRC<br>(optional) | EOF |
|---|---|---|---|---|---|---|
| | 5 bits | 6 bits | 6 bits | $n$ bits | 16 bits | |

**Table 19 — Inventory request format (mask length 16 slots)**

| SOF | Flags | Command | Mask length, $n$<br>16 slots $0 \leqslant n < 44$ | Mask value | CRC<br>(optional) | EOF |
|---|---|---|---|---|---|---|
| | 5 bits | 6 bits | 6 bits | $n$ bits | 16 bits | |

To switch to the next slot, the transceiver sends an EOF.

## 9.3 Request processing by the transponder

Upon reception of a valid request, the transponder shall process it by executing the operation sequence specified in italics as below. The step sequence is also graphically represented in Figure 2.

*NbS is the total number of slots (1 or 16)*

*SN is the current slot number (0 to 15)*

*SN_Length is set to 0 when 1 slot is used and set to 4 when 16 slots are used*

*LSB (value, n) function returns the n least significant bits of value*

*"&" is the concatenation operator*

*Slot_Frame is either a SOF or an EOF*

*SN = 0*

*if NOS flag then*

*NbS = 1 SN_length = 0*

*else NbS = 16 SN_length = 4*

*end if*

*label1: if LSB(UID, SN_length + Mask_length) = LSB(SN, SN_length) & LSB(Mask, Mask_length) then*

*transmit response to inventory request*

*end if*

*wait (Slot_Frame)*

*if Slot_Frame = SOF then*

*stop anticollision and decode/process request*

*exit*

*end if*

*if SN<NbS-1 then*

*SN = SN +1*

*go to label1*

*exit*

*end if*

*exit*

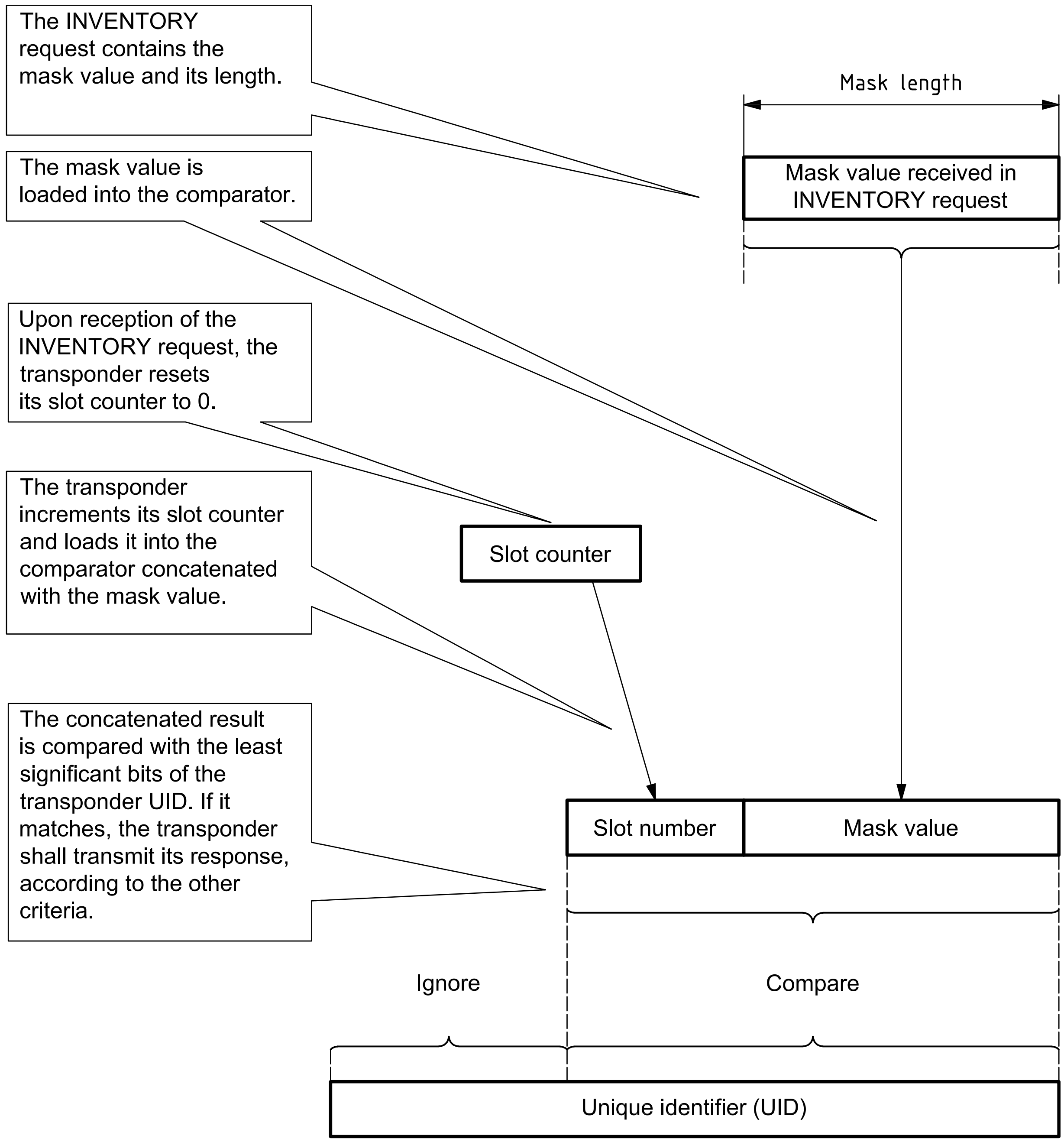



**Figure 2 — Principle of comparison between mask value, slot number and UID**

NOTE When the number of slots is one (NOS flag set to 1), the comparison is made only on the mask.

## 9.4 Explanation of anticollision sequences

### 9.4.1 General

FDX transponders shall transmit the remaining section of the UID in dual pattern code. The following data (error flag, data block where an ISO 11785 number is requested, optional CRC) shall be transmitted in Manchester code.

### 9.4.2 Anticollision sequence with one slot

The following description explains a typical anticollision sequence where the number of slots is 1.

a) The transceiver sends an INVENTORY request.

   If the UID of the transponder is completely unknown, the value of the mask length is set to 0 and the mask value is omitted. After a precisely defined time, all transponders in the ready state simultaneously transmit their responses.

   If the least significant part of the transponder UID is partly known, the attached parameters consist of mask length, $n$, and of the mask value. After a precisely defined time, all transponders in the ready state that have the least significant part of their UID equal to the mask value sent in the INVENTORY request simultaneously transmit their responses.

b) The transceiver checks the transponder responses bitwise.

   If there is no transponder responding, continue at a).

   If there is only one transponder responding, no collision occurs and the transponder UID is received and registered by the transceiver. Continue at c).

   If there is more than one transponder responding, the transceiver reads additional UID bits of the transponders and expands the mask value with these bits until the first collision occurs. The transceiver recognizes the bit position of this collision and expands the mask value to 0 or 1, depending on which serial number branch is to be selected. Continue at a).

c) The transceiver can communicate with the respective transponder by sending requests issued to that transponder. If the transceiver sends another INVENTORY request, continue at a).

### 9.4.3 Anticollision sequence with 16 slots

Figure 3 summarises the main cases that can occur during a typical anticollision sequence where the number of slots is 16.

The different steps are the following.

a) The transceiver sends an INVENTORY request, in a frame, terminated by an EOF. The number of slots is 16.

b) Transponder 1 transmits its response in slot 0. It is the only one to do so, consequently, no collision occurs and its UID is received and registered by the transceiver.

c) The transceiver sends an EOF or recharge for HDX-ADV, meaning to switch to the next slot.

d) In slot 1, two transponders, 2 and 3, transmit their responses, this generates a collision. The transceiver detects it and remembers that a collision was detected in slot 1.

e) The transceiver sends an EOF or recharge for HDX-ADV, meaning to switch to the next slot.

f) In slot 2, no transponder transmits a response. Consequently, the transceiver does not detect a transponder SOF and decides to send an addressed request (for example, a read block) to transponder 1, which UID was already correctly received.

g) All transponders detect a SOF and exit the anticollision sequence. They process this request and since the request is addressed to transponder 1, only transponder 1 transmits its response.

h) All transponders are ready to receive another request. If it is an INVENTORY command, the slot numbering sequence restarts from 0 (slot numbering 0 to 15).

NOTE The decision to interrupt the anticollision sequence is up to the transceiver. It could have continued to send EOFs until slot 15 and then send the request to transponder 1.

### 9.4.4 Mixed population with transponders of type FDX-ADV and HDX-ADV

The following description explains a typical anticollision sequence when transponders of both types (FDX and HDX) are in the transceiver field (or expected to be).

a) The transceiver switches on the RF field and awaits the power-up time.

b) The transceiver performs an anticollision sequence according to 9.4.2 (one slot) or 9.4.3 (16 slots).

c) The transceiver may switch off the RF field.

d) The transceiver switches on the RF field and charges the transponder during 10 ms to 50 ms.

e) The transceiver performs an anticollision sequence according to 9.4.2 (one slot) or 9.4.3 (16 slots).

f) The transceiver switches off the RF field.

NOTE The order of sequence a), b), c), d), e), f) can be replaced by d), e), f), a), b), c).

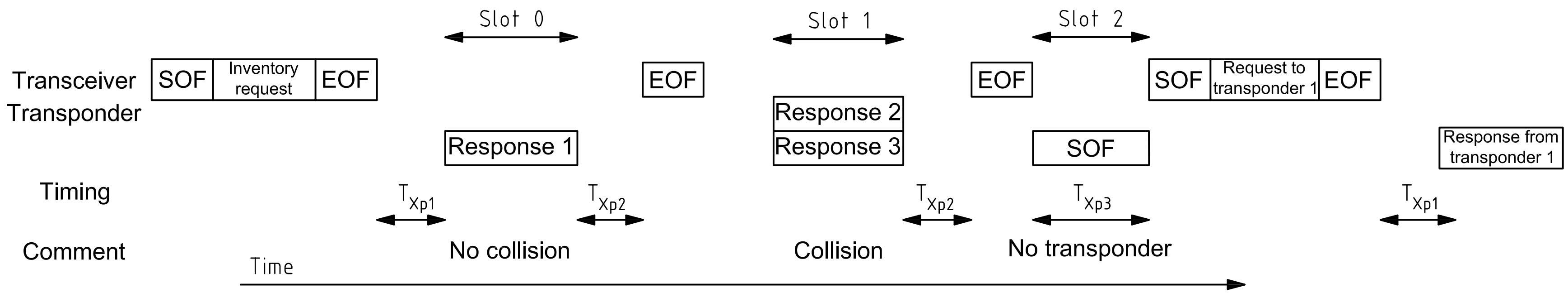
### 9.4.5 Advanced anticollision mode

This mode is used to access the code conforming to ISO 11784 and ISO 11785 during the anticollision process. It can be accessed by setting the inventory flag in multiple block command.

Parameters are:

— read multiple block command with parameter block 0 to 3;

— inventory flag set to 1.




**Key**

$T_{Xp1}$ transponder waiting time before starting to transmit response after detection of valid transceiver request

$T_{Xp2}$ transceiver waiting time before starting to transmit subsequent request after receiving transponder response

$T_{Xp3}$ transceiver waiting time before switching to next slot during inventory process

**Figure 3 — Description of a possible anticollision sequence**

# 10 Commands

## 10.1 Command classification

Four sets of commands are defined as follows (see Table 20).

a) **Mandatory**

All transponders and transceivers shall support them.

b) **Optional**

Transponders may support them, at their option. If not supported, the transponder shall remain silent. If supported, request and response formats shall comply with the definition given in this part of ISO 14223.

c) **Custom**

Transponders support them, at their option, to implement manufacturer specific functions. The function of flags (including reserved bits) shall not be modified. The only fields that can be customized are the parameters and the data fields. If not supported, a correct error code (“Not supported”) or, if no specific error code is supported, error code ‘7’, shall be returned.

d) **Proprietary**

These commands are used by IC and transponder manufacturers for different purposes, such as testing and the programming of system information. They are not specified in this part of ISO 14223. The IC manufacturer may, at his option, document them or not. It is likely that these commands will be disabled after IC and/or transponder manufacturing.

**Table 20 — Command classes**

| Code | Class |
|---|---|
| “00” to “27” | Mandatory or optional |
| “28” to “37” | Custom |
| “38” to “3F” | Proprietary |

All transponders with the same MFC and ICR shall behave in the same manner.

In the case of the implementation of a custom command having a logical function already covered by one of the defined optional commands, it is mandatory that the optional command concerned be implemented.

The attention of transceiver designers is drawn to the possibility that transponder manufacturers can implement custom commands and/or proprietary commands, if not disabled, in quite different ways for the same command code, which can lead to errors whose consequences cannot be predicted. It is therefore recommended that custom commands and/or proprietary commands, if not disabled, be performed only after the IC manufacturer code and IC version has been requested from the transponders. These two parameters, linked with the IC manufacturer information, will inform the transceiver of the supported commands and their syntax.

## 10.2 Command list

Table 21 gives an overview of the command list.

**Table 21 — Command list**

| Command | Code | Type | Function | Valid in state |
|---|---|---|---|---|
| Inventory | “00” | Mandatory | Anticollision loop | Ready |
| Stay quiet | “01” | Mandatory | Forces a transponder into the quiet state | Ready, quiet, selected |
| Read UID | “02” | Mandatory | Fast reading of the transponders UID without collision protection | Ready |
| RFU | “03”-“0F” | — | — | — |
| Read single block | “10” | Optional | Reads a single user memory block | Ready, quiet, selected |
| Read single block with security status | “11” | Optional | Reads a single user memory block with security status | Ready, quiet, selected |
| Read multiple blocks | “12” | Mandatory | Reads multiple user memory blocks | Ready, quiet, selected |
| Read multiple blocks with security status | “13” | Optional | Reads multiple user memory blocks with security status | Ready, quiet, selected |
| Write single block | “14” | Mandatory | Writes a single user memory block | Ready, quiet, selected |
| Write multiple blocks | “15” | Optional | Writes multiple user memory blocks | Ready, quiet, selected |
| Lock block | “16” | Mandatory | Locks a single user memory block | Ready, quiet, selected |
| Get system information | “17” | Optional | Reads specified system memory data | Ready, quiet, selected |
| Select | “18” | Optional | Forces a transponder into the selected state | Ready, quiet, selected |
| Reset to ready | “19” | Optional | Forces a selected transponder into the ready state | Ready, quiet, selected |
| Write system data | “1A” | Optional | Writes specified system data (e.g. DSFID) | Ready, quiet, selected |
| Lock system data | “1B” | Optional | Locks specified system data (e.g. DSFID) | Ready, quiet, selected |
| Reserved (ISO/IEC 18000-2) | “1C” | Not available in ISO 14223 | — | — |
| Read extended multiple blocks | “1D” | Optional | Reads multiple user memory blocks in pages 1 to 255 | Ready, quiet, selected |
| Write extended multiple blocks | “1E” | Optional | Write multiple user memory blocks in pages 1 to 255 | Ready, quiet, selected |
| Lock extended block | “1F” | Optional | Lock user memory block in pages 1 to 255 | Ready, quiet, selected |
| Login | “20” | Optional | RFU | — |
| Change password | “21” | Optional | RFU | — |
| Lock password | “22” | Optional | RFU | — |
| Inventory ISO 11785 code | “23” | Mandatory | Inventory command including ISO 11785 code | Ready |
| RFU | “24” to “27” | Optional | — | — |
| | “28” to “37” | Custom | IC manufacturer specific commands | — |
| | “38” to “3F” | Proprietary | IC manufacturer specific commands | — |

Execution of the lock/write commands shall be completed and acknowledged within 20 ms of the detection of the last falling edge of the transceiver request (FDX) or after the transceiver has switched off the field (HDX).

## 10.3 Mandatory commands

### 10.3.1 INVENTORY

Upon reception of this command without error, all transponders in the ready state shall perform the anticollision sequence. The inventory (INV) flag shall be set to 1. The NOS flag determines whether one or 16 slots are used.

If a transponder detects any error, it shall remain silent.

The INVENTORY request format is given in Table 22, the INVENTORY response format in Table 23, the INVENTORY request format with ISO 11785 code is given in Table 24 and the INVENTORY response format with ISO 11785 code in Table 25.

**Table 22 — INVENTORY request format**

| SOF | Flags | Command<br>INVENTORY | Parameter 1<br>Mask length<br>$n$ | Parameter 2<br>Mask value | CRC<br>(Optional) | EOF |
|---|---|---|---|---|---|---|
| | 5 bits | 6 bits | 6 bits | $n$ bits | 16 bits | |

**Table 23 — INVENTORY response format**

| SOF | Error flag | Data | CRC | EOF |
|---|---|---|---|---|
| | 0 | Remaining section of the UID<br>(UID without mask value) | (Optional) | |
| | 1 bit | 48 – $n$ bits | 16 bits | |

**Table 24 — INVENTORY request format with ISO 11785 code**

| SOF | Flags | Command<br>Inventory<br>ISO 11785 code | Parameter 1<br>Mask length<br>$n$ | Parameter 2<br>Mask value | CRC<br>(Optional) | EOF |
|---|---|---|---|---|---|---|
| | 5 bits | 6 bits | 6 bits | $n$ bits | 16 bits | |

**Table 25 — INVENTORY response format with ISO 11785 code**

| SOF | Error flag | Data | Data block | CRC | EOF |
|---|---|---|---|---|---|
| | 0 | Remaining section of the UID<br>(UID without mask value) | ISO 11785 number | (Optional) | |
| | 1 bit | 48 – $n$ bits | 64 bits | 16 bits | |

The allowed values of $n$ depend on the number of slots and are defined in 9.2.

### 10.3.2 READ UID

Upon reception of this command without error, all transponders in the ready state shall send their UID. The READ UID request format is shown in Table 26 and the READ UID response format in Table 27.

The inventory (INV), addressed (ADR) and select (SEL) flags shall be set to 0.

**Table 26 — READ UID request format**

| SOF | Flags | Command | CRC | EOF |
|---|---|---|---|---|
| | | READ UID | (Optional) | |
| | 5 bits | 6 bits | 16 bits | |

**Table 27 — READ UID response format**

| SOF | Error flag | Data | CRC | EOF |
|---|---|---|---|---|
| | 0 | UID | (Optional) | |
| | 1 bit | 48 bits | 16 bits | |

### 10.3.3 READ MULTIPLE BLOCKS

Upon reception of this command without error, the transponder shall read the requested block(s) and send back their value in the response. The blocks are numbered from 0 to 255. The format for the READ MULTIPLE BLOCKS request is given in Table 28 and the format for the READ MULTIPLE BLOCKS response in Table 29.

The number of blocks in the request is 1 less than the number of blocks that the transponder shall return in its response. For example, a value of 6 in the “Number of blocks” field is needed to request that seven blocks be read, and a value of 0 to request that a single block be read.

**Table 28 — READ MULTIPLE BLOCKS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | | READ MULTIPLE BLOCKS | UID (Optional) | First block number | Number of blocks | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 16 bits | |

**Table 29 — READ MULTIPLE BLOCKS response format**

| SOF | Error flag | Data | CRC | EOF |
|---|---|---|---|---|
| | 0 | User memory block data | (Optional) | |
| | 1 bit | 32 bits | 16 bits | |
| | | Repeated as needed | | |

### 10.3.4 STAY QUIET

Upon reception of this command without error, a transponder in either the ready or the selected state shall enter the quiet state and shall NOT send back a response.

The STAY QUIET command with both SEL and ADR flag set to 0 or both set to 1 is not allowed.

There is *no* response to the STAY QUIET request, even if the transponder detects an error.

The format of the STAY QUIET request format is presented in Table 30.

**Table 30 — STAY QUIET request format**

| SOF | Flags | Command | Parameter | CRC | EOF |
|---|---|---|---|---|---|
| | | STAY QUIET | UID (Optional) | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 16 bits | |

### 10.3.5 WRITE SINGLE BLOCK

Upon reception of this command without error, the transponder shall write the requested block with the data contained in the request and report the success of the operation in the response. The WRITE SINGLE BLOCK request is presented in Table 31 and the WRITE SINGLE BLOCK response in Table 32.

**Table 31 — WRITE SINGLE BLOCK request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Data | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | | WRITE SINGLE BLOCK | UID (Optional) | Block number | Block data | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 32 bits | 16 bits | |

**Table 32 — WRITE SINGLE BLOCK response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

### 10.3.6 LOCK BLOCK

Upon reception of the LOCK BLOCK command without error, the transponder shall lock permanently the requested block. The LOCK BLOCK request format is given in Table 33 and the LOCK BLOCK response format in Table 34.

**Table 33 — LOCK BLOCK request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | CRC | EOF |
|---|---|---|---|---|---|---|
| | | LOCK BLOCK | UID (Optional) | Block number | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 16 bits | |

**Table 34 — LOCK BLOCK response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

## 10.4 Optional commands

### 10.4.1 READ SINGLE BLOCK

Upon reception of this command without error, a transponder shall respond with the content of the respective user memory block.

The READ SINGLE BLOCK request format is presented in Table 35 and the READ SINGLE BLOCK response format in Table 36.

**Table 35 — READ SINGLE BLOCK request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | CRC | EOF |
|---|---|---|---|---|---|---|
| | | READ SINGLE BLOCK | UID (Optional) | Block address | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 16 bits | |

**Table 36 — READ SINGLE BLOCK response format**

| SOF | Error flag | Data | CRC | EOF |
|---|---|---|---|---|
| | 0 | User memory block data | (Optional) | |
| | 1 bit | 32 bits | 16 bits | |

### 10.4.2 READ SINGLE BLOCK WITH SECURITY STATUS

Upon reception of this command without error, the transponder shall read the requested block and the block security status and send back their value in the response.

The READ SINGLE BLOCK WITH SECURITY STATUS request format is presented in Table 37 and the READ SINGLE BLOCK WITH SECURITY STATUS response format in Table 38.

**Table 37 — READ SINGLE BLOCK WITH SECURITY STATUS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | CRC | EOF |
|---|---|---|---|---|---|---|
| | | READ SINGLE BLOCK WITH SECURITY STATUS | UID (Optional) | Block address | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 16 bits | |

**Table 38 — READ SINGLE BLOCK WITH SECURITY STATUS response format**

| SOF | Error flag | Data1 | Data2 | CRC | EOF |
|---|---|---|---|---|---|
| | 0 | Security status | User memory block data | (Optional) | |
| | 1 bit | 4 bits | 32 bits | 16 bits | |

### 10.4.3 READ MULTIPLE BLOCKS WITH SECURITY STATUS

Upon reception of this command without error, the transponder shall read the requested block(s) and the block(s) security status and send back their value in the response, sequentially, block by block. The blocks are numbered from 0 to 255.

The number of blocks in the request is one less than the number of blocks that the transponder is to return in its response. For example, a value of 6 in the "Number of blocks" field is needed to request that seven blocks be read, and a value of 0 to request that a single block be read.

The READ MULTIPLE BLOCKS WITH SECURITY STATUS request format is presented in Table 39 and the READ MULTIPLE BLOCKS WITH SECURITY STATUS response format in Table 40.

**Table 39 — READ MULTIPLE BLOCKS WITH SECURITY STATUS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | | READ MULTIPLE BLOCK WITH SECURITY STATUS | UID (Optional) | First block number | Number of blocks | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 16 bits | |

**Table 40 — READ MULTIPLE BLOCKS WITH SECURITY STATUS response format**

| SOF | Error flag | Data 1 | Data 2 | CRC | EOF |
|---|---|---|---|---|---|
| | 0 | Security status | User memory block data | (Optional) | |
| | 1 bit | 4 bits | 32 bits | 16 bits | |
| | | Repeated as needed | | | |

### 10.4.4 WRITE MULTIPLE BLOCKS

Upon reception of this command without error, the transponder shall write the requested block(s) with the data contained in the request and report the success of the operation in the response.

For HDX transponders, the transceiver shall keep the field on after the EOF for a period sufficient for the memory programming to be performed, so that the transponder is powered.

The blocks are numbered from 0 to 255. The number of blocks in the request is one less than the number of blocks that the transponder is to write. For example, a value of 6 in the "Number of blocks" field is needed to request that seven blocks be written. The "Data" field shall contain seven blocks. A value of 0 in the "Number of blocks" field is needed to request that a single block be written. The "Data" field shall contain one block.

The WRITE MULTIPLE BLOCKS request format is presented in Table 41 and the WRITE MULTIPLE BLOCKS response format in Table 42.

**Table 41 — WRITE MULTIPLE BLOCKS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | Data | CRC | EOF |
|---|---|---|---|---|---|---|---|---|
| | | WRITE MULTIPLE BLOCKS | UID (Optional) | First block number | Number of blocks | Block data | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 32 bits | 16 bits | |
| | | | | | | Repeated as needed | | |

**Table 42 — WRITE MULTIPLE BLOCKS response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

### 10.4.5 GET SYSTEM INFORMATION

Upon reception of this command without error, the transponder shall read the requested system memory block(s) and send back their value in the response.

The GET SYSTEM INFORMATION request format is presented in Table 43, the GET SYSTEM INFORMATION response format in Table 44 and the system data description in Table 45.

**Table 43 — GET SYSTEM INFORMATION request format**

| SOF | Flags | Command | Parameter 1 | CRC | EOF |
|---|---|---|---|---|---|
| | | GET SYSTEM INFORMATION | UID (Optional) | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 16 bits | |

**Table 44 — GET SYSTEM INFORMATION response format**

| SOF | Error flag | Data | | | | | | | | | CRC | EOF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | System memory block data | | | | | | | | | (Optional) | |
| | 1 bit | 40 bits | 8 bits | 8 bits | 8 bits | 8 bits | 8 bits | 8 bits | 8 bits | 8 bits | 16 bits | |
| | | MSN | MFC | ICR | RFU | NSS | NOB | NOP | DSFID | RFU | | |

**Table 45 — System data description**

| System data | Size | Type | Description |
|---|---|---|---|
| MFC | 8 bits | Mandatory | IC manufacturer code |
| ICR | 8 bits | Optional | IC reference code |
| MSN | 40 bits | Mandatory | Manufacturer serial number |
| DSFID | 8 bits | Optional | Data storage format identifier |
| NOP | 8 bits | Optional | Number of pages – 1 (0 to 255) |
| NOB | 8 bits | Optional | Number of blocks per page – 1 (0 to 255) |
| NSS | 8 bits | Optional | Number of sensors |

### 10.4.6 SELECT

The SELECT command shall always be executed with the SEL flag set to 0 and the ADR flag set to 1. Upon reception of this command without error,

a) if the UID is equal to its own UID, the transponder shall enter the selected state and shall send a response, or

b) if it is different,

   — a transponder in a non-selected state (quiet or ready) shall keep its state and shall not send a response, and

   — the transponder in the selected state shall enter the quiet state and shall not send a response.

The SELECT request format is presented in Table 46 and the SELECT response format in Table 47.

**Table 46 — SELECT request format**

| SOF | Flags | Command | Parameter | CRC | EOF |
|---|---|---|---|---|---|
| | | SELECT | UID | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 16 bits | |

**Table 47 — SELECT response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

#### 10.4.7 RESET TO READY

Upon reception of the RESET TO READY command without error, a transponder in the quiet or selected state shall enter the ready state.

If the command is executed in the addressed mode, it shall send a response. If it is executed in *non*-addressed mode, no response shall be sent.

The RESET TO READY request format is presented in Table 48 and the RESET TO READY response format in Table 49.

In the quiet state,

a) if the UID is attached in the command, only one transponder moves from quiet to ready state, and transmits a response, or

b) if no UID is attached to the command, all transponders moves from Quiet state to Ready state without sending a response.

**Table 48 — RESET TO READY request format**

| SOF | Flags | Command | Parameter | CRC | EOF |
|---|---|---|---|---|---|
| | | RESET TO READY | UID (Optional) | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 16 bits | |

**Table 49 — RESET TO READY response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

#### 10.4.8 WRITE SYSTEM DATA

Upon reception of this command without error, the transponder shall write the DSFID value (depending on system data selector, see Table 51) into its memory and report the success of the operation in the response.

The WRITE SYSTEM DATA request format is presented in Table 50 and the WRITE SYSTEM DATA response format in Table 52.

**Table 50 — WRITE SYSTEM DATA request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | | WRITE SYSTEM DATA | UID (Optional) | System data selector[a] | System data | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 2 bits | 8 bits | 16 bits | |

[a] See Table 51.

**Table 51 — System data selector**

| System data selector | System data |
|---|---|
| 00 | RFU |
| 01 | DSFID |
| 10 | RFU |
| 11 | RFU |

**Table 52 — WRITE SYSTEM DATA response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

### 10.4.9 LOCK SYSTEM DATA

Upon reception of this command without error, the transponder shall lock the DSFID value (depending on system data selector, see Table 51) permanently into its memory.

The LOCKSYSTEM DATA request format is presented in Table 53 and the LOCK SYSTEM DATA response format in Table 54.

**Table 53 — LOCK SYSTEM DATA request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | CRC | EOF |
|---|---|---|---|---|---|---|
| | | LOCK SYSTEM DATA | UID (Optional) | System data selector[a] | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 2 bits | 16 bits | |

[a] See Table 51.

**Table 54 — LOCK SYSTEM DATA response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

### 10.4.10 READ EXTENDED MULTIPLE BLOCKS

Upon reception of this command without error, the transponder shall read the requested block(s) from the page specified in parameter 2 of the request and send back their data in the response. The blocks and pages are numbered from 0 to 255.

The number of blocks in the request is one less than the number of blocks that the transponder is to return in its response. For example, a value of 6 in the “Number of blocks” field is needed to request that seven blocks be read. A value of 0 is needed to request that a single block be read.

The READ EXTENDED MULTIPLE BLOCKS request format is presented in Table 55 and the READ EXTENDED MULTIPLE BLOCKS response format in Table 56.

**Table 55 — READ EXTENDED MULTIPLE BLOCKS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | Parameter 4 | CRC | EOF |
|---|---|---|---|---|---|---|---|---|
| | | READ EXT MULTIPLE BLOCKS | UID (Optional) | Page number | First block number | Number of blocks | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 8 bits | 16 bits | |

**Table 56 — READ EXTENDED MULTIPLE BLOCKS response format**

| SOF | Error flag | Data | CRC | EOF |
|---|---|---|---|---|
| | 0 | User memory block data from page number specified in parameter 2 | (Optional) | |
| | 1 bit | 32 bits | 16 bits | |
| | | Repeated as needed | | |

### 10.4.11 WRITE EXTENDED MULTIPLE BLOCKS

Upon reception of this command without error, the transponder shall write the requested blocks within the page specified in parameter 2, with the data contained in the request, and report the success of the operation in the response.

The blocks and pages are numbered from 0 to 255.

The blocks are numbered from 0 to 255. The number of blocks in the request is one less than the number of blocks that the transponder is to write. For example, a value of 6 in the “Number of blocks” field is needed to request that seven blocks be written. The “Data” field shall contain seven blocks. A value of 0 in the “Number of blocks” field is needed to request that a single block be written. The “Data” field shall contain one block.

The WRITE EXTENDED MULTIPLE BLOCKS request format is presented in Table 57 and the WRITE EXTENDED MULTIPLE BLOCKS response format in Table 58.

**Table 57 — WRITE EXTENDED MULTIPLE BLOCKS request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | Parameter 4 | Data | CRC | EOF |
|---|---|---|---|---|---|---|---|---|---|
| | | WRITE EXT MULTIPLE BLOCKS | UID (Optional) | Page number | First block number | Number of blocks | Block data | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 8 bits | 32 bits | 16 bits | |
| | | | | | | | Repeated as needed | | |

**Table 58 — WRITE EXTENDED MULTIPLE BLOCKS response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

The number of blocks may be implementation-dependent.

### 10.4.12 LOCK EXTENDED BLOCK

Upon reception of this command without error, the transponder shall lock the requested block within the page specified in parameter 2 and report the success of the operation in the response.

The blocks and pages are numbered from 0 to 255.

The LOCK EXTENDED BLOCK request format is presented in Table 59 and the LOCK EXTENDED MULTIPLE BLOCKS response format is presented in Table 60.

**Table 59 — LOCK EXTENDED BLOCK request format**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | | LOCK EXT BLOCK | UID (Optional) | Page number | block number | (Optional) | |
| | 5 bits | 6 bits | 48 bits | 8 bits | 8 bits | 16 bits | |

**Table 60 — LOCK EXTENDED BLOCK response format**

| SOF | Error flag | CRC | EOF |
|---|---|---|---|
| | 0 | (Optional) | |
| | 1 bit | 16 bits | |

### 10.4.13 Optional command execution in inventory mode

Some commands may be executed in inventory mode by setting the inventory flag to 1. The support of this mechanism by the transponder is optional.

The list of command codes that can be executed in inventory mode is specified in Table 61. When receiving a request with the inventory flag set to 1, the transponder shall perform the inventory sequence. The inventory mode-related fields that are “Mask Length” and “Mask Value”, followed by the requested command-related parameters (in non-addressed mode), shall be contained in the request (see Table 62).

If the transponder detects an error during the inventory sequence, it shall remain silent.

The syntax of the returned data in the response shall be according to the command code (see Table 63).

FDX transponders transmit the remaining section of the UID in dual pattern code. The following data (error flag, data 2, optional CRC in no error case; error flag, error code, optional CRC in error case) is transmitted in Manchester code.

**Table 61— Commands executable in inventory mode**

| Command | Code | Function |
|---|---|---|
| Read single block | 10 | Reads a single user memory block |
| Read single block with security status | 11 | Reads a single user memory block with security status |
| Read multiple blocks | 12 | Reads multiple user memory blocks |
| Read multiple blocks with security status | 13 | Reads multiple user memory blocks with security status |
| Get system information | 17 | Reads specified system memory data |

**Table 62— Request format of commands executed in inventory mode**

| SOF | Flags | Command | Parameter 1 | Parameter 2 | Parameter 3 | CRC | EOF |
|---|---|---|---|---|---|---|---|
| | 01xxx | [a] | Mask length $n$ | Mask value | Command parameter | (Optional) | |
| | 5 bits | 6 bits | 6 bits | $n$ bits | See relevant command | 16 bits | |

[a] See Table 61.

**Table 63 — Response format to command in inventory mode**

| SOF | Error Flag | Data 1 | Data 2 | CRC | EOF |
|---|---|---|---|---|---|
| | 0 | Remaining section of the UID (UID without Mask value) | Response Data as defined in the executed command | (Optional) | |
| | 1 bit | 48 – $n$ bits | xx bits | 16 bits | |

The allowed values of $n$ depend on the number of slots and are defined in 9.2.

## 10.5 Custom commands

This part of ISO 14223 does not specify custom commands by definition.

## 10.6 Proprietary commands

This part of ISO 14223 does not specify proprietary commands by definition.

# Annex A
(informative)

# Description of a typical anticollision sequence with FDX and HDX transponders

Table A.1 shows an example of a mixed population with two transponders of FDX (FDX1, FDX2) and two transponders of HDX (HDX1, HDX2). The transceiver performs the anticollision sequence row by row, whereby the decision to start with FDX or HDX is with the transceiver.

In the example, the transceiver chooses the slot number to 1. The transceiver detects a collision and decides to proceed with the transponders having a 0 at the UID collision position. After recognizing the complete UID of the first transponder (FDX1, HDX1), the transceiver proceeds with the transponders having a 1 at the UID collision position (FDX2, HDX2).

The timings are specified in Clause 8.

**Table A.1 — Anticollision sequence example of a mixed population with two FDX transponders ( FDX1, FDX2) and two HDX transponders (HDX1, HDX2)**

| Performer | FDX-B | HDX-Adv |
|---|---|---|
| Transceiver | Field ON | Field ON |
| Transceiver | Wait for power on time (min. 2,5 ms) | Charge transponder for max. 50 ms |
| Transceiver | INVENTORY request<br>(NOS = 1, no mask value) | INVENTORY request<br>(NOS = 1, no mask value) |
| Transceiver | — | Field OFF |
| Transponder | Wait t1 (max. 1,7 ms) | Wait for max. 2 ms |
| Transponder | Response of UID (with collision) | Response of UID (with collision) |
| Transceiver | Wait t2 (min. 1,2 ms) | Field ON with recharge for max. 10 ms |
| Transceiver | INVENTORY request<br>(NOS = 1, mask value + 0) | INVENTORY request<br>(NOS = 1, mask value + 0) |
| Transceiver | — | Field OFF |
| Transponder | Wait t1 (max. 1,7 ms) | Wait for max. 2 ms |
| Transponder | Response of remaining UID [A1] | Response of remaining UID [B1] |
| Transceiver | Wait t2 (min. 1,2 ms) | Field ON with recharge for max. 10 ms |
| Transceiver | INVENTORY request<br>(NOS = 1, mask value + 1) | INVENTORY request<br>(NOS = 1, mask value + 1) |
| Transceiver | — | Field OFF |
| Transponder | Wait t1 (max. 1,7 ms) | Wait for max. 2 ms |
| Transponder | Response of remaining UID [A2] | Response of remaining UID [B2] |
| Transceiver | Field OFF | Field OFF |
| NOTE All times are worst case times. | | |

# Bibliography

[1] ISO/IEC 18000-2, *Information technology — Radio frequency identification for item management — Part 2: Parameters for air interface communications below 135 kHz*

ICS 35.240.99; 65.020.30

Price based on 34 pages