

---

---

**Information and documentation —  
RFID in libraries —**

**Part 3:  
Fixed length encoding**

*Information et documentation — RFID dans les bibliothèques —  
Partie 3: Encodage de longueur fixe*





**COPYRIGHT PROTECTED DOCUMENT**

© ISO 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
CP 401 • Ch. de Blandonnet 8  
CH-1214 Vernier, Geneva  
Phone: +41 22 749 01 11  
Email: [copyright@iso.org](mailto:copyright@iso.org)  
Website: [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

Page

<b>Foreword</b>	<b>v</b>
<b>Introduction</b>	<b>vi</b>
<b>1 Scope</b>	<b>1</b>
<b>2 Normative references</b>	<b>1</b>
<b>3 Terms and definitions</b>	<b>1</b>
<b>4 Requirements</b>	<b>3</b>
4.1 Data elements	3
4.2 RFID air interface	3
4.2.1 Air interface conformance	3
4.2.2 Tag performance	3
4.3 Data protocol	3
<b>5 General encoding rules</b>	<b>3</b>
5.1 Distinguishing from other applications and encodings	3
5.2 Writing/reading direction	3
5.3 Memory area layout	4
5.3.1 Specifications	4
5.3.2 Layout for tags greater than 32 bytes	4
5.3.3 Layout for 32-byte tags	4
5.4 Strings and integers	4
5.4.1 String encoding	4
5.4.2 Integer encoding	5
5.5 Writing the tag	5
5.5.1 Cyclic redundancy check (CRC)	5
5.5.2 Unused space	5
5.5.3 End of tag	5
5.6 Reading optimization	5
5.7 Profiling	5
5.8 Locking	5
<b>6 Data elements</b>	<b>5</b>
<b>7 Data blocks</b>	<b>9</b>
7.1 Types of data blocks	9
7.2 Basic block	9
7.3 Special blocks	11
7.4 Structured extension blocks	11
7.4.1 Usage of structured extension blocks	11
7.4.2 Format of structured extension blocks	12
7.4.3 Length	12
7.4.4 Data block ID	12
7.4.5 Checksum	12
7.5 Library extension block	13
7.6 Acquisition extension block	13
7.7 Library supplement block	14
7.8 Title block	14
7.9 ILL block	15
7.10 Unstructured extension blocks	15
7.10.1 Usage of unstructured extension blocks	15
7.10.2 Format of unstructured extension blocks	16
<b>8 Miscellaneous</b>	<b>16</b>
8.1 Migration	16
<b>Annex A (informative) Information about ISO 28560 RFID in libraries</b>	<b>17</b>

**Annex B (informative) Encoding examples ..... 18**

**Annex C (informative) Cyclic redundancy check (CRC) ..... 22**

**Annex D (informative) Reading optimization ..... 23**

**Annex E (informative) Guidelines for regional profiling ..... 24**

**Bibliography ..... 25**

## 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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://www.iso.org/directives)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](http://www.iso.org/patents)).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html).

This document was prepared by Technical Committee ISO/TC 46, *Information and documentation*, Subcommittee SC 4, *Technical interoperability*.

This third edition cancels and replaces the second edition (ISO 28560-3:2014), of which it constitutes a minor revision.

A list of all parts in the ISO 28560 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](http://www.iso.org/members.html).

## **Introduction**

Libraries are implementing radio frequency identification (RFID) as item identification to replace bar codes. RFID streamlines applications like user self-service, security, and materials handling. This standard data model for encoding information on RFID tags increases the cost-effectiveness of the technology within libraries, particularly through greater interoperability of RFID tags and equipment, and enhance support for resource sharing between libraries.

This document deals with the encoding of a basic set of data elements in a fixed length format and the rest of the data elements in optional extension blocks. ISO 28560-1 defines the set of mandatory and optional data elements.

ISO 28560-2 and this document are mutually exclusive with respect to an RFID tag being applied to a loan item. In other words, the RFID tag is encoded according to the rules of this document, or to the rules of ISO 28560-2, or to some proprietary rules. Depending on the technologies being used, and other features of tags that are claiming compliance with ISO 28560-2, the reading system might achieve a degree of interoperability.

This document provides essential standards-based information about RFID in libraries. Ongoing advice needs to be provided because of the evolving nature of RFID technology, and the opportunities to migrate between different types of legacy system and encoding rules of this document.

# Information and documentation — RFID in libraries —

## Part 3: Fixed length encoding

### 1 Scope

This document provides a data model and encoding rules for the use of radio frequency identification (RFID) tags for items appropriate for the needs of all types of libraries (including national, academic, public, corporate, special, and school libraries).

This document specifies the rules for encoding

- a subset of data elements taken from the total set of data elements listed in ISO 28560-1 into a basic block, and
- other data elements into extension blocks onto the RFID tag.

A source of additional information about implementation issues is provided in [Annex A](#).

### 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 28560-1, *Information and documentation — RFID in libraries — Data elements and general guidelines for implementation*

ISO/IEC 10646, *Information technology — Universal coded character set (UCS)*

ISO/IEC 18000-3, *Information technology — Radio frequency identification for item management — Part 3: Parameters for air interface communications at 13,56 MHz*

ISO/IEC 18046-3, *Information technology — Radio frequency identification device performance test methods — Part 3: Test methods for tag performance*

ISO/IEC 18047-3, *Information technology — Radio frequency identification device conformance test methods — Part 3: Test methods for air interface communications at 13,56 MHz*

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 28560-1 and the following apply.

#### 3.1

##### **basic block**

data block occupying the first 272 bits of the RFID tag

Note 1 to entry: If the RFID tag is limited to 256 bits (i.e. 32 bytes), the basic block is truncated.

**3.2**

**byte**

**8-bit byte**

group of eight consecutive bits

Note 1 to entry: A byte can represent one *character* ([3.3](#)) or be part of a representation of a character.

**3.3**

**character**

one or more *bytes* ([3.2](#))

**3.4**

**CRC**

**cyclic redundancy check**

value calculated from the data on the tag

**3.5**

**data block**

container for encoding data elements, *CRC* ([3.4](#)), filler, and end mark

**3.6**

**end block**

*data block* ([3.5](#)) containing the end mark terminating the information on the RFID tag

**3.7**

**extension block**

optional *data block* ([3.5](#)) following the *basic block* ([3.1](#))

**3.8**

**field**

entry in a *data block* ([3.5](#))

**3.9**

**filler data block**

optional *data block* ([3.5](#)) that can be inserted to align other data blocks on *page* ([3.11](#)) boundaries

**3.10**

**fixed length field**

*field* ([3.8](#)) of prescribed size in a *data block* ([3.5](#))

**3.11**

**page**

minimum data unit that can be read from or written to a tag

Note 1 to entry: This is measured in *bytes* ([3.2](#)).

**3.12**

**string**

sequence of *characters* ([3.3](#))

**3.13**

**unsigned integer**

binary value of a number of consecutive bits

**3.14**

**variable length field**

*field* ([3.8](#)) of variable size in a *data block* ([3.5](#))



## 4 Requirements

### 4.1 Data elements

The data elements shall be as defined and compliant with those listed in ISO 28560-1.

NOTE There is a degree of flexibility in using locally defined codes that enable enhancements and variations to be implemented while still complying with the basic set of data elements.

### 4.2 RFID air interface

#### 4.2.1 Air interface conformance

The air interface for compliant tags shall be in accordance with the specification for Mode 1.

For migration purposes, additional non-compliant air interfaces used in legacy systems can be supported during a transition period, which is permitted to remain in place for years, as necessary.

The air interface conformance shall be tested in accordance with ISO/IEC 18047-3.

#### 4.2.2 Tag performance

Where there are requirements for test tag performance, these shall be performed in accordance with ISO/IEC 18046-3.

### 4.3 Data protocol

The fixed length encoding described in this document does not require a separate data protocol.

## 5 General encoding rules

### 5.1 Distinguishing from other applications and encodings

The value of the application family identifier (AFI) is used to distinguish tags for library applications from other applications. The values of AFI for library applications are defined in ISO 28560-1.

ISO 28560-1 describes how the data storage format identifier (DSFID), if present in the system memory as a programmable register, is used to distinguish tags in the library application area, i.e. with the same AFI.

Tags encoded according to this document shall be programmed with the value  $3E_{\text{HEX}}$  in the DSFID register if the tag contains a programmable DSFID register.

This document is not able to encode the DSFID if the tag does not contain a programmable DSFID register. In this case, ISO 28560-2 encodes the DSFID in the first byte of the working area of the tag. To take this situation into account, the content parameter (see [Table 1](#)) shall not take the value 6 on RFID tags encoded according to this document.

If and only if the tag does not contain a programmable DSFID register, it is permissible to distinguish tags encoded according to this document from other encodings by verifying the cyclic redundancy check (CRC) encoded in the basic block (see [7.2](#)).

### 5.2 Writing/reading direction

Data shall be written to and read from the tag as specified in ISO/IEC 18000-3, Mode 1, such that the first bit transmitted to or from the tag is the least significant bit of the first field of the basic block. This field contains the content parameter (see [Table 1](#)). From that starting point, bytes are transmitted to

or from the tag in order from left to right, with byte 0 to the left of bytes 1, 2, and 3, as shown in the memory map in [Annex B](#).

## 5.3 Memory area layout

### 5.3.1 Specifications

The memory area shall be encoded starting with a basic block with fixed length encoding of a basic set of data elements for use in the library.

If the size of the RFID tag is limited to 256 bits (32 bytes), the RFID tag can only contain a truncated basic block.

If the size of the RFID tag is greater than 256 bits, extension blocks (structured or unstructured) can be inserted after the basic block up to the capacity of the chip. If extension blocks are inserted, the order of these is optional. The length of an extension block is determined by the first byte of the block. The type of extension block is defined in the following two bytes.

Filler data blocks may be inserted between blocks to align to page boundaries.

An end block shall terminate the encoding, unless the basic block and possible extension blocks take up the whole space on the RFID tag, in which case an end block is not needed.

### 5.3.2 Layout for tags greater than 32 bytes

The layout for tags greater than 32 bytes (256 bits) shall be as follows:

<basic block>[(<filler data block>)\*<extension block>]\*(<filler data block>)\*(<end block>)

The end block is mandatory if the tag is not full (see [5.3.1](#)). Basic block, filler data block, structured extension blocks, unstructured extension blocks, and end block are specified in [Clause 7](#).

An example is given in [Annex B](#).

### 5.3.3 Layout for 32-byte tags

The layout for 32-byte tags shall be as follows:

<truncated basic block>

The truncated basic block is specified in [Clause 7](#).

An example is given in [Annex B](#).

## 5.4 Strings and integers

### 5.4.1 String encoding

All strings shall be encoded in UTF-8 in accordance with ISO/IEC 10646, with the first character of the string stored in the lowest memory location. Note that UTF-8 encoding implies that a character can occupy more than one byte.

The end of a string can be defined in the following different ways:

- with one byte 00<sub>HEX</sub>;
- with the length of a fixed length field;
- with the end of a structured extension block.

For fixed length fields, all unused bytes shall be 00<sub>HEX</sub>.

For variable length fields, one byte 00<sub>HEX</sub> shall be used between each field.

#### 5.4.2 Integer encoding

Integer-encoded fields shall use 4, 8, or 16 bits unsigned integers.

### 5.5 Writing the tag

#### 5.5.1 Cyclic redundancy check (CRC)

For RFID tags with only 32 bytes (256 bits) user data space of the basic block is truncated two bytes, but the CRC shall be calculated for a full-length basic block with the two missing bytes assumed to be 00<sub>HEX</sub>.

See [7.2](#) and [Annex C](#) for a description of CRC.

#### 5.5.2 Unused space

Any unused space in blocks shall be filled with 00<sub>HEX</sub>, i.e. a 6-byte primary item identifier encoded in the basic block (see [7.2](#)) shall be written as the 6-byte primary item identifier followed by 10 bytes 00<sub>HEX</sub>. It is very important for reading optimization that this rule be followed.

#### 5.5.3 End of tag

If a data block ends on the last user byte of a tag, no “end block” is required. The length specified in the last data block shall not indicate a size larger than the tag.

### 5.6 Reading optimization

Guidelines for reading optimization are given in [Annex D](#).

### 5.7 Profiling

Guidelines for regional profiling are given in [Annex E](#).

### 5.8 Locking

It is technically possible to lock parts of the tag, but this document does not prescribe any strategy for locking. Such a strategy is left for regional profiling.

## 6 Data elements

[Table 1](#) shows for each data element defined in ISO 28560-1 the data block where it is encoded, how it is encoded, and the values it can take. Note that some data elements can be encoded in different data blocks. The data blocks are described in [Clause 7](#).

Table 1 — Data elements

<i>N</i> <sup>a</sup>	Name of data element <sup>b</sup>	Data block <sup>c</sup>	Encoding <sup>d</sup>	Values <sup>e</sup>	Requirements and remarks <sup>f</sup>
1	Primary item identifier	Basic block or library extension block	If the primary item identifier is maximum 16 bytes, it shall be encoded in the basic block as a string.  Otherwise, it shall be encoded as a string in the library extension block.	Any string	If a primary identifier is not assigned yet, the string is empty.
2	Content parameter	Basic block	4-bit unsigned integer	1 (14 values are reserved for future use: 0, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15. To be able to distinguish from tags encoded according to ISO 28560-2, the value 6 shall not be used.)	The value defines a version number.  A new version number shall be applied if and only if it refers to a new version of this part of ISO 28560, which is not backward compatible.
3	Owner institution (ISIL)	Basic block or library extension block	If the ISIL prefix is one or two characters and the ISIL unit identifier is less than or equal to 11 bytes (or 9 bytes for 32 bytes tags), the string can be encoded in the basic block. In this case, the string shall be formed as the concatenation of the prefix and the unit identifier. If the prefix is only one character, a blank is added between the prefix and the unit identifier.  If the prefix is more than two characters or the unit identifier is longer than 11 bytes, the string can be encoded in the library extension block. In this case, the string shall be the ISIL code, including the hyphen.	If the tag is limited to 32 bytes, only ISIL codes with a prefix of a maximum of two characters and a unit identifier less than 9 bytes can be encoded.  If the size of the tag is more than 32 bytes, any ISIL code can, in theory, be encoded.	The ISIL code is defined in ISO 15511 as <prefix><hyphen><unit identifier>, where <prefix> shall be either an alpha-2 country code (two upper-case letters), or another registered string (non-country code), and where <unit identifier> can be up to 11 characters long.

<sup>a</sup> This column specifies the data element number, i.e. the number identifying the data element in ISO 28560-1. Data elements 14, and 27 to 31 are reserved for future use, and therefore not listed in [Table 1](#). When usage and type for these data elements are defined, a mapping should be considered and specified, and then published at the informational website in [Annex A](#).

<sup>b</sup> This column specifies the data element name, i.e. the name identifying the data element in ISO 28560-1.

<sup>c</sup> This column specifies the names of the data blocks in which the data element is encoded.

<sup>d</sup> This column specifies how the data element is encoded. For encoding of strings and integers, see [5.4.1](#) and [5.4.2](#).

<sup>e</sup> This column specifies the values the data element can take.

<sup>f</sup> This column gives additional requirements and remarks.

Table 1 (continued)

<i>N</i> <sup>a</sup>	Name of data element <sup>b</sup>	Data block <sup>c</sup>	Encoding <sup>d</sup>	Values <sup>e</sup>	Requirements and remarks <sup>f</sup>
4	Set information	Basic block	Two 8-bit unsigned integers, the first specifying <number of parts in item> and the second <ordinal part number>.	<numbers of parts in item> can take the values 0 to 255, where 0 indicates an unspecified number. <ordinal part number> can take the values 0 to 255, where 0 indicates the first item in a set, where not all items have an RFID tag.	Set information consists of two parts: <number of parts in item> and <ordinal part number> See examples in ISO 28560-1.
5	Type of usage	Basic block or library extension block	Main qualifier shall be encoded in basic block as a 4-bit unsigned integer.  As a supplement, the whole data element can be encoded in the library extension block as an 8-bit unsigned integer.	The set of code values is specified in ISO 28560-1.	The data element consists of two parts: <main qualifier> and <sub-qualifier>.
6	Shelf location	Library supplement block	String	Any string specifying the location of the item	See ISO 28560-1.
7	ONIX media format	Library supplement block	String	ONIX media descriptor of two alphabetic characters	See ISO 28560-1.
8	MARC media format	Library supplement block	String	MARC 21 category of material descriptor	See ISO 28560-1.
9	Supplier identifier	Acquisition extension block	String	Any string identifying the supplier of the item	See ISO 28560-1.
10	Order number	Acquisition extension block	String	Any string identifying the order containing the item	See ISO 28560-1.
11	ILL borrowing institution (ISIL)	ILL block	String	Any ISIL code in the form <prefix><hyphen><unit identifier>	The ISIL code is defined in ISO 15511.
12	ILL borrowing transaction number	ILL block	String	Any string identifying the borrowing transaction	See ISO 28560-1.

<sup>a</sup> This column specifies the data element number, i.e. the number identifying the data element in ISO 28560-1. Data elements 14, and 27 to 31 are reserved for future use, and therefore not listed in [Table 1](#). When usage and type for these data elements are defined, a mapping should be considered and specified, and then published at the informational website in [Annex A](#).

<sup>b</sup> This column specifies the data element name, i.e. the name identifying the data element in ISO 28560-1.

<sup>c</sup> This column specifies the names of the data blocks in which the data element is encoded.

<sup>d</sup> This column specifies how the data element is encoded. For encoding of strings and integers, see [5.4.1](#) and [5.4.2](#).

<sup>e</sup> This column specifies the values the data element can take.

<sup>f</sup> This column gives additional requirements and remarks.

**Table 1** (continued)

<i>N</i> <sup>a</sup>	Name of data element <sup>b</sup>	Data block <sup>c</sup>	Encoding <sup>d</sup>	Values <sup>e</sup>	Requirements and remarks <sup>f</sup>
13	GS1 product identifier	Acquisition extension block	String	Any GTIN-13 code	See ISO 28560-1.
15	Local data A	Unstructured extension blocks	Locally defined	Locally defined	See ISO 28560-1.
16	Local data B	Unstructured extension blocks	Locally defined	Locally defined	See ISO 28560-1.
17	Title	Title block	String	Any string specifying the titles of the item	See ISO 28560-1.
18	Product identifier local	Acquisition extension block	String	Any string specifying the product identifier of the item	See ISO 28560-1.
19	Media format (other)	Library extension block	8-bit unsigned integer	0: Undefined 1: Book 2: CD/DVD 3: Magnetic tape 4: Other 5: Other, careful handling is required 6: Very small item, special handling is required	See ISO 28560-1.
20	Supply chain stage	Acquisition extension block	8-bit unsigned integer	The set of code values are specified in ISO 28560-1.	See ISO 28560-1.
21	Supplier invoice number	Acquisition extension block	String	Any string identifying the invoice containing the item	See ISO 28560-1.
22	Alternative item identifier	Library extension block	String	Any string identifying the item	See ISO 28560-1.

<sup>a</sup> This column specifies the data element number, i.e. the number identifying the data element in ISO 28560-1. Data elements 14, and 27 to 31 are reserved for future use, and therefore not listed in [Table 1](#). When usage and type for these data elements are defined, a mapping should be considered and specified, and then published at the informational website in [Annex A](#).

<sup>b</sup> This column specifies the data element name, i.e. the name identifying the data element in ISO 28560-1.

<sup>c</sup> This column specifies the names of the data blocks in which the data element is encoded.

<sup>d</sup> This column specifies how the data element is encoded. For encoding of strings and integers, see [5.4.1](#) and [5.4.2](#).

<sup>e</sup> This column specifies the values the data element can take.

<sup>f</sup> This column gives additional requirements and remarks.

Table 1 (continued)

<i>N</i> <sup>a</sup>	Name of data element <sup>b</sup>	Data block <sup>c</sup>	Encoding <sup>d</sup>	Values <sup>e</sup>	Requirements and remarks <sup>f</sup>
23	Alternative owner institution	Basic block or library extension block	If the non-ISIL code is less than or equal to 10 bytes (or 8 bytes for 32 bytes tags), it should be encoded as a string in the basic block.  Otherwise, it may be encoded as a string in the library extension block.	Any string identifying the institution that owns the item, that is either a national standardized code not part of ISIL or a code that is neither part of ISIL nor a national standard	See ISO 28560-1.
24	Subsidiary of an owner institution	Library supplement block	String	Any string defined by the institution that owns the item	See ISO 28560-1.
25	Alternative ILL borrowing institution	ILL block	String	Any string identifying the institution that borrows the item	See ISO 28560-1.
26	Local data C	Unstructured extension blocks	Locally defined	Locally defined	See ISO 28560-1.
<p><sup>a</sup> This column specifies the data element number, i.e. the number identifying the data element in ISO 28560-1. Data elements 14, and 27 to 31 are reserved for future use, and therefore not listed in <a href="#">Table 1</a>. When usage and type for these data elements are defined, a mapping should be considered and specified, and then published at the informational website in <a href="#">Annex A</a>.</p> <p><sup>b</sup> This column specifies the data element name, i.e. the name identifying the data element in ISO 28560-1.</p> <p><sup>c</sup> This column specifies the names of the data blocks in which the data element is encoded.</p> <p><sup>d</sup> This column specifies how the data element is encoded. For encoding of strings and integers, see <a href="#">5.4.1</a> and <a href="#">5.4.2</a>.</p> <p><sup>e</sup> This column specifies the values the data element can take.</p> <p><sup>f</sup> This column gives additional requirements and remarks.</p>					

## 7 Data blocks

### 7.1 Types of data blocks

The following types of data blocks exist:

- basic block;
- special blocks;
- structured extension blocks;
- unstructured extension blocks.

The rules for encoding these different types of data blocks are given in [7.2](#) to [7.10](#).

### 7.2 Basic block

The basic block contains a number of fixed length data fields. The basic block occupies the first 34 bytes (272 bits) on the tag. If the tag has only 32 bytes (256 bits), the layout for the truncated basic block shall be used. In this case, no other data can be stored on the tag.

[Table 2](#) and [Table 3](#) specify the usage and layout of the basic block and the truncated basic block.



**Table 2 — Usage and layout of the basic block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0 (0)	4 bits	Content parameter (data element 2)
0 (4)	4 bits	Type of usage <main qualifier> (data element 5)
1	2 bytes	Set information (data element 4) <numbers of parts in item> and <ordinal part number>, the first stored in the low byte and the second in the next byte
3	16 bytes	This field can contain the empty string or the primary item identifier (data element 1) if it is maximum 16 bytes. Otherwise, the first byte shall be 01 <sub>HEX</sub> , and the primary item identifier shall be encoded in the library extension block.
19	2 bytes	This field shall contain the CRC, which is not a data element but a value calculated as part of the encoding. The purpose is to detect reading and writing errors. CRC-16-CCITT shall be used with $x^{16} + x^{12} + x^5 + 1$ polynomial and FFFF as start value. The CRC shall be calculated starting from the lowest address and skipping the two CRC bytes.
21	13 bytes	This field can contain the empty string or the owner institution (ISIL) (data element 3), in which case the first two bytes shall be a two-character country code or a one-letter non-country code followed by a blank. The following bytes shall be the unit identifier. The country code, the non-country code, and the unit identifier are defined in ISO 15511. The hyphen specified in ISO 15511 shall not be encoded. If the third byte is 01 <sub>HEX</sub> , the owner institution (ISIL) (data element 3) shall be encoded in the library extension block; in this case, the value of the other bytes in the field is undefined. If the third byte is 02 <sub>HEX</sub> or 03 <sub>HEX</sub> , the string starting from byte 4 shall be the alternative owner institution (data element 23); in this case, the value of the first two bytes in the field is undefined. 02 <sub>HEX</sub> shall indicate a national standardized code not part of ISIL and 03 <sub>HEX</sub> shall indicate an institution code that is neither part of ISIL nor a national standard. Use of ISIL is highly recommended. The escape options should only be used in a transition period or if ISIL is irrelevant.
<sup>a</sup> Start position of the field given as byte number, with bit number in parentheses. If the bit number is not given, 0 is assumed.		
<sup>b</sup> Length of field.		
<sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> .		

**Table 3 — Usage and layout of the truncated basic block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0 (0)	4 bits	Content parameter (data element 2)
0 (4)	4 bits	Type of usage <main qualifier> (data element 5)
1	2 bytes	Set information (data element 4) <numbers of parts in item> and <ordinal part number>, the first stored in the low byte and the second in the next byte
3	16 bytes	This field can contain the empty string or the primary item identifier (data element 1) if it is maximum 16 bytes. Otherwise, the first byte is 01 <sub>HEX</sub> , and the primary item identifier shall be encoded in the library extension block.
<sup>a</sup> Start position of the field given as byte number with bit number in parentheses. If the bit number is not given, 0 is assumed.		
<sup>b</sup> Length of field measured in bytes or bits.		
<sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> .		



**Table 3 (continued)**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
19	2 bytes	<p>This field shall contain the CRC, which is not a data element but a value calculated as part of the encoding. The purpose is to detect reading and writing errors.</p> <p>CRC-16-CCITT shall be used with <math>x^{16} + x^{12} + x^5 + 1</math> polynomial and FFFF as start value. The CRC shall be calculated starting from the lowest address and skipping the two CRC bytes.</p> <p>See special remark in the next row.</p>
21	11 bytes	<p>This field can contain the empty string or the owner institution (ISIL) (data element 3), in which case the first two bytes shall be a two-character country code or a one-letter non-country code followed by a blank. The following bytes shall be the unit identifier. The country code, the non-country code, and the unit identifier are defined in ISO 15511. The hyphen specified in ISO 15511 shall not be encoded.</p> <p>If the third byte is 01<sub>HEX</sub>, the owner institution (ISIL) (data element 3) shall be encoded in the library extension block; in this case, the value of the other bytes in the field is undefined.</p> <p>If the third byte is 02<sub>HEX</sub> or 03<sub>HEX</sub>, the string starting from byte 4 shall be the alternative owner institution (data element 23); in this case the value of the first two bytes in the field is undefined. 02<sub>HEX</sub> shall indicate a national standardized code not part of ISIL and 03<sub>HEX</sub> shall indicate an institution code that is neither part of ISIL nor a national standard.</p> <p>Use of ISIL is highly recommended. The escape option should only be used in a transition period or if ISIL is irrelevant.</p> <p>Special remark:</p> <p>For CRC calculation this field shall be viewed as being 13 bytes long, i.e. the CRC calculation algorithm shall add two bytes 00<sub>HEX</sub> to the end of the field.</p>
<p><sup>a</sup> Start position of the field given as byte number with bit number in parentheses. If the bit number is not given, 0 is assumed.</p> <p><sup>b</sup> Length of field measured in bytes or bits.</p> <p><sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a>.</p>		

### 7.3 Special blocks

The layout specified in [5.3](#) operates with two special blocks, each 1 byte of size.

[Table 4](#) specifies the usage and layout of these special blocks.

**Table 4 — Usage and layout of the special blocks**

Block name	Description	Encoding
End block	If there is room left on the tag, this block shall be added to signal the end of the data.	1 byte = 00 <sub>HEX</sub>
Filler data block	The block may be used to align other data blocks on page boundaries. If necessary multiple instances may be used.	1 byte = 01 <sub>HEX</sub>

### 7.4 Structured extension blocks

#### 7.4.1 Usage of structured extension blocks

Structured extension blocks are used as a supplement to the basic block to encode the full set of data elements specified in ISO 28560-1.

It is permissible to have two or more structured extension blocks of the same type (i.e. with the same data block ID).

#### 7.4.2 Format of structured extension blocks

Structured extension blocks use a frame structure of 4 bytes to specify length, type and checksum. The general structure of the extension blocks is:

<length><data block ID><XOR checksum>(<data field><end block>)\*

Data fields have to occur in the order specified for the specific structured extension block. A data field can be empty. In this case, it shall be filled with 00<sub>HEX</sub>.

It is permissible to limit the structured extension block size by any length, in which case data fields that reach beyond this limit can be truncated or omitted. If so, the missing data are assumed to be bytes with value 00<sub>HEX</sub>.

It is also permissible to specify a larger size than the data field requires. If so, the redundant part of the structured extension block is filled with 00<sub>HEX</sub>.

#### 7.4.3 Length

The length specifies the number of bytes of the extension block. The value shall be greater than 4 and shall specify all bytes in the extension block, including the length byte itself.

The length is encoded as an 8-bit unsigned integer.

#### 7.4.4 Data block ID

The data block ID identifies the structured extension block.

The data block ID is encoded as a 16-bit unsigned integer with least significant byte (lsb) stored at the lowest memory location.

Values:

- 1: Acquisition extension block
- 2: Acquisition extension block
- 3: Library supplement block
- 4: Title block
- 5: ILL block
- 6 to 100: Other structured extension blocks (for future use)

#### 7.4.5 Checksum

##### 7.4.5.1 General

The checksum is a value calculated from the bytes in the extension block and used to verify the encoding.

##### 7.4.5.2 Calculating the checksum

Set the checksum field to 00<sub>HEX</sub>, then calculate the XOR for all bytes including length, ID, and checksum. Store this value in the checksum field.

##### 7.4.5.3 Verifying the checksum

Calculate the XOR for all bytes including length, ID, and checksum field. The result shall be 00<sub>HEX</sub>.

## 7.5 Library extension block

This block contains four data fields. It can be used as a supplement to the basic block for specification of media format and for item and institution identification.

[Table 5](#) specifies the usage and layout of the library extension block.

**Table 5 — Usage and layout of the library extension block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0	1	Length
1	2	Data block ID (=1)
3	1	XOR checksum
4	1	Media format (other) (data element 19).
5	Variable	This field can be empty or it can contain the primary item identifier (data element 1), or the alternative item identifier (data element 22). Please note that the primary item identifier should be stored in the basic block if possible.
First free byte <sup>d</sup>	Variable	<p>This field can be empty or it contain the owner institution (ISIL) (data element 3), or the alternative owner institution (data element 23). Please note that these data elements should be stored in the basic block if possible.</p> <p>The owner institution (ISIL) shall be encoded as a string according to ISO 15511 (ISIL), i.e. &lt;prefix&gt;&lt;hyphen&gt;&lt;unit identifier&gt;</p> <p>The alternative owner institution shall be encoded as a string preceded by 02<sub>HEX</sub> (specifying a national standardized code not part of ISIL) or 03<sub>HEX</sub> (specifying a code that is neither part of ISIL nor a national standard).</p>
First free byte <sup>d</sup>	1	Type of usage (data element 5)
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field. <sup>d</sup> First byte after the terminating 00 <sub>HEX</sub> of the preceding variable length field.		

## 7.6 Acquisition extension block

This block contains six data fields. It can be used to hold information relevant for the acquisition phase.

[Table 6](#) specifies the usage and layout of the acquisition extension block.

**Table 6 — Usage and layout of the acquisition extension block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0	1	Length
1	2	Data block ID (=2)
3	1	XOR checksum
4	Variable	This field can be empty or it can contain the supplier identifier (data element 9).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the product identifier local (data element 18).
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field. <sup>d</sup> First byte after the terminating 00 <sub>HEX</sub> of the preceding variable length field.		

**Table 6 (continued)**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the order number (data element 10).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the supplier invoice number (data element 21).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the GS1 product identifier (data element 13).
First free byte <sup>d</sup>	1	Supply chain stage (data element 20)
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field. <sup>d</sup> First byte after the terminating 00 <sub>HEX</sub> of the preceding variable length field.		

## 7.7 Library supplement block

This block contains four data fields. It may be used to hold additional information on an item.

[Table 7](#) specifies the usage and layout of the library supplement block.

**Table 7 — Usage and layout of the library supplement block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0	1	Length
1	2	Data block ID (=3)
3	1	XOR checksum
4	Variable	This field can be empty or it can contain the shelf location (data element 6).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the MARC media format (data element 8).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the ONIX media format (data element 7).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the subsidiary of an owner institution (data element 24).
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field. <sup>d</sup> First byte after the terminating 00 <sub>HEX</sub> of the preceding variable length field.		

## 7.8 Title block

This block contains one data field. It can be used to hold title information to an item.

[Table 8](#) specifies the usage and layout of the title block.

**Table 8 — Usage and layout of the title block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0	1	Length
1	2	Data block ID (=4)
3	1	XOR checksum
4	Variable	This field may be empty or it may contain the title (data element 17).
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field.		

## 7.9 ILL block

This block contains three data fields. It can be used to hold information relevant for interlibrary loan.

[Table 9](#) specifies the usage and layout of the ILL block.

**Table 9 — Usage and layout of the ILL block**

Position <sup>a</sup>	Length <sup>b</sup>	Content of field <sup>c</sup>
0	1	Length
1	2	Data block ID (=5)
3	1	XOR checksum
4	Variable	This field can be empty or it can contain the ILL borrowing institution (ISIL) (data element 11). The ILL borrowing institution (ISIL) shall be encoded as a string according to ISO 15511 (ISIL), i.e. <prefix><hyphen><unit identifier>
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the ILL borrowing transaction number (data element 12).
First free byte <sup>d</sup>	Variable	This field can be empty or it can contain the alternative ILL borrowing institution (data element 25). The alternative ILL borrowing institution shall be encoded as a string preceded by 02 <sub>HEX</sub> (specifying a national standardized code not part of ISIL) or 03 <sub>HEX</sub> (specifying a code that is neither part of ISIL nor a national standard).
<sup>a</sup> Start position of the field given as byte number. <sup>b</sup> Length of field measured in bytes. <sup>c</sup> Content of field. For data elements, refer to <a href="#">Table 1</a> and for string and integer encoding, refer to <a href="#">5.4</a> . Note that for variable length fields, one byte 00 <sub>HEX</sub> shall be inserted between each field. <sup>d</sup> First byte after the terminating 00 <sub>HEX</sub> of the preceding variable length field.		

## 7.10 Unstructured extension blocks

### 7.10.1 Usage of unstructured extension blocks

For these blocks, it is up to the local or national level to profile the selection and order of local data elements and other data elements.

Local data elements from ISO 28560-1:

Local data A

Local data B

Local data C

Other data elements not part of ISO 28560-1 can be specified on a local or national level.

#### **7.10.2 Format of unstructured extension blocks**

An unstructured extension block shall be identified by a unique data block ID greater than 100 to distinguish the blocks from the structured extension blocks (see [7.4.4](#)).

Encoding is defined on a local or national level.

## **8 Miscellaneous**

### **8.1 Migration**

The decision to migrate from a legacy implementation to a data model based on this document depends on economic and operational considerations that are beyond the scope of this document.

## **Annex A**

### **(informative)**

## **Information about ISO 28560 RFID in libraries**

### **A.1 Informational website**

Additional information about RFID in libraries: <http://biblstandard.dk/rfid> hosted by Royal Danish Library.

### **A.2 Types of support information**

At the time of publication of this document, two items of information have been identified. These are listed below, together with the URL of the continually updated material.

- RFID in libraries. Links to external materials: <http://biblstandard.dk/rfid/docs/RFID-in-libraries-Links-external>
- RFID in libraries. Q&A: <http://biblstandard.dk/rfid/docs/RFID-in-libraries-q-and-a>

Other material can be published in the future, and this can be publicized and made available from the URL in [A.1](#).

## Annex B (informative)

### Encoding examples

#### B.1 Example 1, encoding of truncated basic block

[Table B.1](#) gives an example of encoding of a truncated basic block (see [7.2](#)).

The truncated mode is used for tags with only 256 bits (32 bytes).

[Table B.1](#) gives an example with specific values of a set of data elements to be encoded in a truncated basic block. The data elements and the indexes in the content column refer to [Table 1](#). The length and the encoding columns indicate the space occupied by and the hexadecimal value of the data elements for the given value.

**Table B.1 — Example 1, data elements to be encoded**

Content	Length	Value	Encoding (hex value)
2. Content parameter	4 bits	1	1
5. Type of usage	4 bits	1 (item for circulation)	1
4. Set information	2 bytes	Item 1 of 1	0101
1. Primary item identifier	16 bytes	1000000056	3130303030303030303536000000000000
CRC	2 bytes		98A4
3. Owner institution (ISIL)	11 bytes	DK-718500	4444B3731383530300000000

The example given in [Table B.1](#) results in the memory map for the encoded tag, shown in the shaded area in [Table B.2](#).

The page number column identifies the given page of the tag. The byte number column identifies the byte location on the tag. The dump column shows a graphical representation of the byte values.

**Table B.2 — Example 1, memory map**

Page number	Byte number	Byte value hex	Byte value hex	Byte value hex	Byte value hex	Dump
0	0 to 3	11	01	01	31	... 1
1	4 to 7	30	30	30	30	0 0 0 0
2	8 to 11	30	30	30	35	0 0 0 5
3	12 to 15	36	00	00	00	6 ...
4	16 to 19	00	00	00	98	... "
5	20 to 23	A4	44	4B	37	¤ D K 7
6	24 to 27	31	38	35	30	1 8 5 0
7	28 to 31	30	00	00	00	0 ...

#### B.2 Example 2, encoding of basic block and structured extension blocks

The example shows encoding of a basic block, a library extension block, and an acquisition extension block on a tag.



The basic block is described in 7.2. Structured extension blocks are described in 7.4.

Table B.3 gives an example with specific values of a set of data elements to be encoded in a truncated basic block and two extension blocks. The data elements and the indexes in the content column refer to Table 1. The length and the encoding columns indicate the space occupied by and the hexadecimal value of the data elements for the given value.

The example requires a tag with at least 608 bits (76 bytes). In this case, the basic block is 272 bits (34 bytes).

**Table B.3 — Example 2, data elements to be encoded**

Code	Content	Length	Value hex	Encoding (hex value)
B	2. Content parameter	4 bits	1	1
B	5. Type of usage	4 bits	1 (item for circulation)	1
B	4. Set information	2 bytes	Item 1 of 1	0101
B	1. Primary item identifier	16 bytes	1000000136	31303030303030303133360000000000
B	CRC	2 bytes		3615
B	3. Owner institution (ISIL)	13 bytes	DK-718500	444B373138353030000000000000
L	Length	1 byte	5 bytes	05
L	Data block ID	2 bytes	1 (library ext. block)	01 00
L	XOR checksum	1 byte		05
L	19. Media format (other)	1 byte	1 (book)	01
A	Length	1 byte	34 bytes	22
A	Data block ID	2 bytes	2 (acquisition ext. block)	02 00
A	XOR checksum	1 byte		71
A	9. Supplier identifier	10 bytes	Bogvognen	426F67766F676E656E00
A	18. Product identifier local	11 bytes	1234567890	3132333435363738393000
A	10. Order number	1 byte	Empty	00
A	21. Supplier invoice number	9 bytes	A789656c	613738393635366300

The code indicates the block type:  
 B = basic block  
 L = library extension block  
 A = acquisition extension block

The example given in Table B.3 results in the memory map for the encoded tag shown in the shaded area in Table B.4.

The page number column identifies the given page of the tag. The byte number column identifies the byte location on the tag. The dump column shows a graphical representation of the byte values.

**Table B.4 — Example 2, memory map**

Code	Page number	Byte number	Byte value hex	Byte value hex	Byte value hex	Byte value hex	Dump
B	0	0 to 3	11	01	01	31	... 1
B	1	4 to 7	30	30	30	30	0 0 0 0

The code indicates the block type:  
 B = basic block  
 L = library extension block  
 A = acquisition extension block

**Table B.4** (continued)

Code	Page number	Byte number	Byte value hex	Byte value hex	Byte value hex	Byte value hex	Dump
B	2	8 to 11	30	30	30	35	0 0 0 5
B	3	12 to 15	36	00	00	00	6 ...
B	4	16 to 19	00	00	00	98	... "
B	5	20 to 23	A4	44	4B	37	▯ D K 7
B	6	24 to 27	31	38	35	30	1 8 5 0
B	7	28 to 31	30	00	00	00	0 ...
B	8	32 to 33	00	00			..
L	8	34 to 35			05	01	..
L	9	36 to 38	00	05	01		...
A	9	39				22	"
A	10	40 to 43	02	00	71	42	.. q B
A	11	44 to 47	6F	67	76	6F	o g v o
A	12	48 to 51	67	6E	65	6E	g n e n
A	13	52 to 55	00	32	32	33	. 1 2 3
A	14	56 to 59	34	35	36	37	4 5 6 7
A	15	60 to 63	38	39	30	00	8 9 0 .
A	16	64 to 67	00	61	37	38	. a 7 8
A	17	68 to 71	39	36	35	36	9 6 5 6
A	18	72 to 75	63	00	00	00	c ...
The code indicates the block type: B = basic block L = library extension block A = acquisition extension block							

### B.3 Example 3, encoding of Primary item identifier

Primary item identifier is a mandatory data element. [Table 1](#), entry 1 tells that it can be encoded in two different data blocks: either basic block or library extension block.

The choice depends on the length of Primary item identifier. If it is less than or equal to 16 bytes (octets), it shall always be encoded in basic block as specified in [Table 2](#).

Otherwise, byte position 3 in basic block shall be set to 01<sub>HEX</sub> and Primary item identifier shall be encoded in library extension block as specified in [Table 5](#).

If the RFID tag is limited to 32 bytes, it can only contain a truncated basic block (see [Table 3](#)). In this case, only Primary item identifiers less than or equal to 16 bytes (octets) can be encoded on the tag.

### B.4 Example 4, encoding of Owner institution (ISIL)

Owner institution (ISIL) is a strongly recommended data element and should be preferred instead of Alternative owner institution. [Table 1](#), entry 3 tells that it can be encoded in two different data blocks: either basic block or library extension block.

The choice depends on the length of the prefix and the unit identifier of the ISIL code. If the prefix is one or two bytes (octets) and the unit identifier is less than or equal to 11 bytes (octets), the data element shall always be encoded in basic block as specified in [Table 2](#). In this case, the encoded string shall be

formed as the concatenation of the prefix and the unit identifier. If the prefix is only one character, a blank (U+0020) shall be added between the prefix and the unit identifier.

Otherwise, byte position 23 in basic block shall be set to 01<sub>HEX</sub> and the data element shall be encoded in library extension block as specified in [Table 5](#). In this case, the string shall be the ISIL code including the hyphen.

If the RFID tag is limited to 32 bytes, it can only contain a truncated basic block (see [Table 3](#)). In this case, only ISIL with a one- or two-character prefix and a unit identifier less than or equal to 9 bytes (octets) can be encoded on the tag.

**NOTE** ISIL is defined in ISO 15511. It is a variable length identifier that shall consist of a maximum of 16 characters, using digits (Arabic numerals 0 to 9), or letters from the 26-character Roman alphabet (excluding the use of any diacritical marks) and the following special marks: solidus (/), hyphen (-) and colon (:). Each ISIL identifier shall be unique in accordance with the repertoire of characters specified in ISO/IEC 10646 without regard to case.

For each character and special mark in this limited character repertoire, the UTF-8 encoding (which is used for encoding of strings in ISO 28560-3) occupies 1 byte (also called octet).

**EXAMPLE 1** The ISIL identifier “O-FITHE” is encoded in basic block as the string “O FITHE” (Note that the prefix “O” is a short ISIL form of the non-country code “OCLC”; the hyphen is omitted, but a blank is added because the prefix is only one character.)

**EXAMPLE 2** The ISIL identifier “DK-820010” is encoded in basic block as “DK820010”.

**EXAMPLE 3** The hypothetical ISIL identifier “WXYZ-ABCD” is encoded in library extension block as “WXYZ-ABCD”.

**EXAMPLE 4** The hypothetical ISIL identifier “AB-DEFGHIJKLMNOPQRS” is encoded in library extension block as “AB-DEFGHIJKLMNOPQRS”.

## B.5 Example 5, encoding of Alternative owner institution

Alternative owner institution can be used instead of Owner institution (ISIL). [Table 1](#), entry 23 tells that it can be encoded in two different data blocks: either basic block or library extension block.

The choice depends on the length of Alternative owner institution. If it is less than or equal to 10 bytes (octets), it shall always be encoded in basic block as specified in [Table 2](#). In this case, bytes 21 and 22 are undefined and byte 23 is either 02<sub>HEX</sub> or 03<sub>HEX</sub>, where 02<sub>HEX</sub> indicates a national standardized code not part of ISIL and 03<sub>HEX</sub> an institution code that is neither part of ISIL nor a national standard.

Otherwise, byte position 23 in basic block is set to 01<sub>HEX</sub> and Alternative owner institution is encoded in library extension block as specified in [Table 5](#).

If the RFID tag is limited to 32 bytes, it can only contain a truncated basic block (see [Table 3](#)). In this case, only Alternative owner institution less than or equal to 8 bytes (octets) can be encoded on the tag.

## Annex C (informative)

### Cyclic redundancy check (CRC)

#### C.1 Specification

The CRC-16-CCITT shall be used.

#### C.2 Example

The string “RFID tag data model” gives the CRC value 1AEE, that shall be encoded with EE at the lowest memory location and 1A at the next memory location.

#### C.3 Example code

```
const int crc_poly=0x1021;
int crc_sum;
void update_crc(int c)
{
    int i;
    bool xor_flag;
    c<<=8;
    for(i=0; i<8; i++)    {
        xor_flag=((crc_sum ^ c) & 0x8000)!=0;
        crc_sum = crc_sum << 1;
        if (xor_flag) crc_sum = crc_sum ^ crc_poly;
        c = c << 1;
    }
    crc_sum&=0xffff;
}
...
crc_sum=0xffff;
call "update_crc(byte)"; for each data byte
```

## **Annex D**

### **(informative)**

## **Reading optimization**

### **D.1 General**

Any application should be capable of reading any tag conforming to this document.

### **D.2 Fast reading**

When the “primary item identifier” is 12 bytes or less, it is possible to make a faster but less reliable reading. Read the first 16 bytes and verify that the last byte is 00<sub>HEX</sub>.

This way of reading the RFID tag cannot verify the CRC as defined in this document. It is also not possible to verify if the item belongs to the library or if it is a non-library tag.

### **D.3 Optimized reading**

When the owner institution (ISIL) data element is 10 bytes or less, the full basic block header can be read and checked by reading 32 bytes. In this case, it should be verified that the last byte is 00<sub>HEX</sub>.

### **D.4 Structured or unstructured extensions**

Any application should normally be able to handle any item by just reading the basic block. However, an increased speed or service level may be provided if the correct extensions are present.

## **Annex E** **(informative)**

### **Guidelines for regional profiling**

Regional profiles can specify the following:

- that certain structured data blocks be used;
- that certain structured data blocks not be used;
- that certain optional data elements in the basic block or in structured extension blocks be used;
- that certain optional data elements in the basic block or in structured extension blocks not be used;
- that unstructured extension blocks be used;
- that certain data elements require write protection.

## Bibliography

- [1] ISO 15511, *Information and documentation — International standard identifier for libraries and related organizations (ISIL)*

