
**Identification cards — Contactless
integrated circuit cards — Vicinity
cards —**

**Part 3:
Anticollision and transmission protocol**

**AMENDMENT 3: Extended VICC
memory organisation**

*Cartes d'identification — Cartes à circuit(s) intégré(s) sans contact —
Cartes de voisinage —*

Partie 3: Anticollision et protocole de transmission

*AMENDEMENT 3: Organisation de la mémoire étendue des cartes de
voisinage à circuit(s) intégré(s)*

IECNORM.COM : Click to view the full PDF of ISO/IEC 15693-3:2009/AMD3:2015



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, 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
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

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).

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).

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

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: [Foreword - Supplementary information](#)

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and personal identification*.

IECNORM.COM : Click to view the full PDF of ISO/IEC 15693-3:2009/AMD3:2015

Identification cards — Contactless integrated circuit cards — Vicinity cards —

Part 3: Anticollision and transmission protocol

AMENDMENT 3: Extended VICC memory organisation

Page 6, Clause 5

Replace the paragraph with the following:

The commands specified in this part of ISO/IEC 15693 assume that the physical memory is organized in blocks (or pages) of fixed size.

- Up to 65536 blocks can be addressed.
- Block size can be of up to 256 bits.
- This leads to a maximum memory capacity of up to 2 MBytes (16 MBits).

The commands described in this part of ISO/IEC 15693 allow the access (read and write) by block(s). There is no implicit or explicit restriction regarding other access method (e.g. by byte or by logical object in future revision(s) of this part of ISO/IEC 15693 or in custom commands).

Page 22, 10.2

Replace [Table 8](#) with:

Table 8 — Command codes

Command code	Type	Function
'01'	Mandatory	Inventory
'02'	Mandatory	Stay quiet
'03' – '1F'	Mandatory	RFU
'20'	Optional	Read single block
'21'	Optional	Write single block
'22'	Optional	Lock block
'23'	Optional	Read multiple blocks
'24'	Optional	Write multiple blocks
'25'	Optional	Select
'26'	Optional	Reset to ready
'27'	Optional	Write AFI
'28'	Optional	Lock AFI
'29'	Optional	Write DSFID
'2A'	Optional	Lock DSFID
'2B'	Optional	Get system information
'2C'	Optional	Get multiple block security status

Table 8 (continued)

Command code	Type	Function
'30'	Optional	Extended read single block
'31'	Optional	Extended write single block
'32'	Optional	Extended lock block
'33'	Optional	Extended read multiple blocks
'34'	Optional	Extended write multiple blocks
'3C'	Optional	Extended get multiple block security status
'2D' – '2F' '35' – '3B' '3D' – '9F'	Optional	RFU
'A0' – 'DF'	Custom	IC Mfg dependent
'E0' – 'FF'	Proprietary	IC Mfg dependent

Page 36

Add new subclauses 10.4.14, 10.4.15, 10.4.16, 10.4.17, 10.4.18, 10.4.19:

10.4.14 Extended read single block**Command code = '30'**

When receiving the Extended read single block command, the VICC shall read the requested block and send back its value in the response.

If a VICC supports Extended read single block command, it shall also support Read single block command for the first 256 blocks of memory.

If the Option_flag is set in the request, the VICC shall return the block security status, followed by the block value.

If it is not set, the VICC shall return only the block value.

SOF	Flags	Extended read single block	UID	Block number	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	16 bits	

Figure 53 — Extended read single block request format**Request parameter:**

(Optional) UID

Block number

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 54 — Extended read single block response format when Error_flag is set

SOF	Flags	Block security status	Data	CRC16	EOF
	8 bits	8 bits	Block length	16 bits	

Figure 55 — Extended read single block response format when Error_flag is NOT set

Response parameter:

Error_flag (and Error code if Error_flag is set)

if Error_flag is not set

Block security status (if Option_flag is set in the request)

Block data

10.4.15 Extended write single block**Command code = '31'**

When receiving the Extended write single block command, the VICC shall write the requested block with the data contained in the request and report the success of the operation in the response.

If a VICC supports Extended write single block command, it shall also support Write single block command for the first 256 blocks of memory.

Option_flag definition, see ISO/IEC 15693-3:2009/Amd2:20XX, Clause 9.5.

SOF	Flags	Extended write single block	UID	Block number	Data	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	Block length	16 bits	

Figure 56 — Extended write single block request format

Request parameter:

(Optional) UID

Block number

Data

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 57 — Extended write single block response format when Error_flag is set

SOF	Flags	CRC16	EOF
	8 bits	16 bits	

Figure 58 — Extended write single block response format when Error_flag is NOT set

Response parameter:

Error_flag (and Error code if Error_flag is set)

10.4.16 Extended lock block**Command code = '32'**

When receiving the Extended lock block command, the VICC shall lock permanently the requested block.

If a VICC supports Extended lock block command, it shall also support Lock block command for the first 256 blocks of memory.

Option_flag definition, see ISO/IEC 15693-3:2009/Amd2:20XX, Clause 9.5.

SOF	Flags	Extended lock block	UID	Block number	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	16 bits	

Figure 59 — Extended lock single block request format

Request parameter:

(Optional) UID

Block number

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 60 — Extended lock block response format Error_flag is set

SOF	Flags	CRC16	EOF
	8 bits	16 bits	

Figure 61 — Extended lock block response format when Error_flag is NOT set

Response parameter:

Error_flag (and Error code if Error_flag is set)

10.4.17 Extended read multiple blocks**Command code = '33'**

When receiving the Extended read multiple blocks command, the VICC shall read the requested block(s) and send back their value in the response.

If a VICC supports Extended read multiple blocks command, it shall also support Read multiple blocks command for the first 256 blocks of memory.

If the Option_flag is set in the request, the VICC shall return the block security status, followed by the block value sequentially block by block.

If the Option_flag is not set in the request, the VICC shall return only the block value.

The blocks are numbered from '0000' to 'FFFF' (0 to 65535).

The number of blocks in the request is one less than the number of blocks that the VICC shall return in its response.

EXAMPLE A value of '0006' in the "Number of blocks" field requests to read 7 blocks. A value of '0000' requests to read a single block.

SOF	Flags	Extended read multiple block	UID	First block number	Number of blocks	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	16 bits	16 bits	

Figure 62 — Extended read multiple blocks request format

Request parameter:

(Optional) UID

First block number

Number of blocks

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 63 — Extended read multiple blocks response format when Error_flag is set

SOF	Flags	Block security status	Data	CRC16	EOF
	8 bits	8 bits	Block length	16 bits	
Repeated as needed					

Figure 64 — Extended read multiple block response format when Error_flag is NOT set

Response parameter:

Error_flag (and Error code if Error_flag is set)

if Error_flag is not set (the following order shall be respected in the VICC response)

Block security status N (if Option_flag is set in the request)

Block value N

Block security status N+1 (if Option_flag is set in the request)

Block value N+1

etc.

where N is the first requested (and returned) block.

10.4.18 Extended write multiple blocks

Command code = '34'

When receiving the Extended write multiple blocks command, the VICC shall write the requested block(s) with the data contained in the request and report the success of the operation in the response.

If a VICC supports Extended write multiple blocks command, it shall also support Write multiple blocks command for the first 256 blocks of memory.

Option_flag definition, see ISO/IEC 15693-3:2009/Amd2:20XX, Clause 9.5.

The blocks are numbered from '0000' to 'FFFF' (0 to 65535).

The number of blocks in the request is one less than the number of blocks that the VICC shall write.

EXAMPLE A value of '0006' in the "Number of blocks" field requests to write 7 blocks. The "Data" field shall contain 7 blocks. A value of '0000' in the "Number of blocks" field requests to write 1 block. The "Data" field shall contain 1 block.

SOF	Flags	Extended write multiple block	UID	First block number	Number of blocks	Data	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	16 bits	Block length	16 bits	
						Repeated as needed		

Figure 65 — Extended write multiple blocks request format

Request parameter:

(Optional) UID

First block number

Number of blocks

Block data (repeated as defined in [Figure 65](#))

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 66 — Extended write multiple blocks response format when Error_flag is set

SOF	Flags	CRC16	EOF
	8 bits	16 bits	

Figure 67 — Extended write multiple block response format when Error_flag is NOT set

Response parameter:

Error_flag (and Error code if Error_flag is set)

10.4.19 Extended get multiple block security status**Command code = '3C'**

When receiving the Extended get multiple block security status command, the VICC shall send back the block security status.

The blocks are numbered from '0000' to 'FFFF' (0 to 65535).

The number of blocks in the request is one less than the number of block security status that the VICC shall return in its response.

EXAMPLE A value of '0006' in the "Number of blocks" field requests to return 7 Block security status. A value of '0000' in the "Number of blocks" field requests to return a single Block security status.

SOF	Flags	Extended get multiple block security status	UID	First block number	Number of blocks	CRC16	EOF
	8 bits	8 bits	64 bits	16 bits	16 bits	16 bits	

Figure 68 — Extended get multiple block security status request format

Request parameter:

(Optional) UID

First block number

Number of blocks

SOF	Flags	Error code	CRC16	EOF
	8 bits	8 bits	16 bits	

Figure 69 — Extended get multiple block security status response format when Error_flag is set