INTERNATIONAL STANDARD

ISO/IEC 18013-2

Second edition
2020-06

AMENDMENT 1
2023-04

Personal identification — ISO-compliant driving licence —

Part 2: Machine-readable technologies

AMENDMENT 1: DG11 length for compact encoding

Reference number
ISO/IEC 18013-2:2020/Amd. 1:2023(E)

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

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

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 document 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 or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC 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) or the IEC list of patent declarations received (see https://patents.iec.ch).

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. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, SC 17, *Cards and security devices for personal identification*.

A list of all parts in the ISO/IEC 18013 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 and www.iec.ch/national-committees.

Personal identification — ISO-compliant driving licence —

Part 2:
Machine-readable technologies

AMENDMENT 1: DG11 length for compact encoding

Page 4, Clause 3.1.8

Add the following Note

“Note 3 to entry: See Annex B and Annex E (informative)."

Page 24, Clause B.1

Replace

"The annex also prescribes means of validating and authenticating the stored data."

with

"Means for authenticating the stored data are specified in ISO/IEC 18013-3.

This version 02 of compact encoding supersedes the deprecated version 01 included in Annex E (informative) for the information of manufacturers of readers that must be able to read IDL cards issued in accordance with this version that have not yet expired.”

Page 25, Clause B.4.1

Replace

"[header] × [Data Group 1] × [Data Group 2] × [Data Group 3] × [Data Group 4] × [Data Group 7] × [Data Group 11]¶"

with

"[header] × [Data Group 1] × [Data Group 2] × [Data Group 3] × [Data Group 4] × [Data Group 7] × [Data Group 11] × [Data Group 12] × [DG.SOD.1] × [DG.SOD.H]¶"

and insert the following statement:

"Data Group 12, DG.SOD.1, and DG.SOD.H are specified in ISO/IEC 18013-3."

Replace

“Data Group 11 is followed by the end of file delimiter (¶).”

with

“DG.SOD.H is followed by the end of file delimiter (¶).”

Page 25, Clause B.4.2

In definition of Version, replace ‘01’ with ‘02’.

Page 26, Clause B.4.2

In the EXAMPLE replace the WG 10 version level "1" with "2".

In the EXAMPLE replace ‘A0 00 00 02 48 01 00 01 00 82 06 3E’ with ‘A0 00 00 02 48 01 00 02 00 82 06 3E’.

Page 31, Clause B.5.11

Insert the following at the end of the clause:

However, DG11 shall include the DG length.

Data Group 11 thus shall be coded as follows (spaces are included to enhance legibility only, and are not encoded):

[previous data group] × [DG11 length] [DG11 data block] × [next data group]

where

[DG11 length]	is the length of the [DG11 data block] field, expressed using ASN.1 rules;
[DG 11 data block]	is a variable length field, encoded in accordance with Table B.1, with the understanding that delimiters may be included in data fields as data (i.e. without fulfilling a delimiting function).

Page 31, Clause B.5.12

Replace the clause with the following overall example:

Matthias Werner Kleinscmidt (a male) was born on 5 March 1972. On 31 August 2007, the Roads Authority of the Namibia Department for Transport issued Matthias Werner an IDL (number 123456789AB) that expires after 5 years. The issuing authority has also assigned an administrative number (03/7203055367088) to the IDL.

The IDL authorises Matthias Werner to drive Light Vehicles (since 5 August 2003), Medium Sized Goods Vehicles (since 11 August 2005) and Passenger Vehicles (since 2 February 2007).

As domestic data, the IDL includes a TRIPS number (NAM/01/16/1234567/0) and Professional Driving Permit (PrDP) authorisation to drive both Passenger (P) and Goods (G) vehicles until 20 August 2009.

One portrait image (1003 bytes) using JPEG2000 compression is included. The minutiae of his right index finger are stored as well (613 byte data block; compact size, with ridge skeleton end and ridge skeleton bifurcation points).

The above information will logically be written as set out below (square brackets and spaces are used where appropriate to improve readability; single quotes signify hexadecimal numbers):

['A0 00 00 02 48 01 00'] ['02 00'] [82 07 0F] × KLEINSCMIDT ÷ MATTHIAS WERNER ÷ ’19 72 03 05’ ÷ ’20 07 08 31’ ÷ ’20 12 08 30’ ÷ NAM ÷ Roads Authority ÷ 123456789AB ÷ B ; ’20 03 08 05’;;;;; C1 ; ’20 05 04 11’ ;;;;; D ; ’20 07 02 02’ ;;;; × 1 ÷÷÷÷÷ × 03/7203055367088 ÷ ’01’ ÷ ’01’ ÷ ’00 00 00 00’ × ['4'] [1003] [Data Group 4 Portrait Image] × ['01 01'] ['00 06'] [613] [Data Group 7 Fingerprint

template] × [30] NAM/01/16/1234567/0 ÷ ÷ P ; G ;; '20 09 08 30' × × [DG.SOD.1 length] [digital signature] × [DG.SOD.H length] [hash value]¶

The above fields will be encoded as follows (all numbers are hexadecimal):

[A0 00 00 02 48 01 00] [02 00] [82 07 0F] × [4B 4C 45 49 4E 53 43 4D 49 44 54] ÷ [4D 41 54 54 48 49 41 53 20 57 45 52 4E 45 52] ÷ [19 72 03 05] ÷ [20 07 08 31] ÷ [20 12 08 30] ÷ [4E 41 4D] ÷ [52 6F 61 64 73 20 41 75 74 68 6F 72 69 74 79] ÷ [31 32 33 34 35 36 37 38 39 41 42] ÷ [42 3B 20 03 08 05 3B 3B 3B 3B 3B 43 31 3B 20 05 04 11 3B 3B 3B 3B 3B 44 3B 20 07 02 02 3B 3B 3B 3B] × [01] ÷÷÷÷÷ × [30 33 2F 37 32 30 33 30 35 35 33 36 37 30 38 38] ÷ [01] ÷ [01] ÷ [00 00 00 00] × [04] [82 03 EB] [Data Group 4 Portrait Image] × [01 01] [00 06] [82 02 65] [Data Group 7 Fingerprint template] × [1E 4E 41 4D 2F 30 31 2F 31 36 2F 31 32 33 34 35 36 37 2F 30] ÷ ÷ [50 3B 47 3B 3B 20 09 08 30] × × [DG.SOD.1 length] [digital signature] × [DG.SOD.H length] [hash value] ¶

The final byte stream will be as follows:

A0 00 00 02 48 01 00 02 00 82 07 0F D7 4B 4C 45 49 4E 53 43 4D 49 44 54 F7 4D 41 54 54 48 49 41 53 20 57 45 52 4E 45 52 F7 19 72 03 05 F7 20 07 08 31 F7 20 12 08 30 F7 4E 41 4D F7 52 6F 61 64 73 20 41 75 74 68 6F 72 69 74 79 F7 31 32 33 34 35 36 37 38 39 41 42 F7 42 3B 20 03 08 05 3B 3B 3B 3B 3B 43 31 3B 20 05 04 11 3B 3B 3B 3B 3B 44 3B 20 07 02 02 3B 3B 3B 3B D7 01 F7 F7 F7 F7 F7 F7 D7 30 33 2F 37 32 30 33 30 35 35 33 36 37 30 38 38 F7 01 F7 01 F7 00 00 00 00 D7 04 82 03 EB [Data Group 4 Portrait Image] D7 01 01 00 06 82 02 65 [Data Group 7 Fingerprint template] D7 1E 4E 41 4D 2F 30 31 2F 31 36 2F 31 32 33 34 35 36 37 2F 30 F7 F7 50 3B 47 3B 3B 20 09 08 30 D7 D7 [DG.SOD.1 length] [digital signature] D7 [DG.SOD.H length] [hash value] B6

Page 33, Clause B.6

In the EXAMPLE replace 'A0 00 00 02 48 01 00 03 00 82 07 0A' with 'A0 00 00 02 48 03 00 02 00 82 07 0A'.

In the EXAMPLE replace '53' 'A0 00 00 02 48 03 00 01 00 82 07 0A … B6' with '53' 'A0 00 00 02 48 03 00 02 00 82 07 0A … B6'

After Annex D

Add new Annex E.

Annex E
(informative)

Compact encoding (Version 01)

E.1 General

This annex defines the deprecated compact encoding scheme (ver 01) for the information of manufacturers of readers that must be able to read IDL cards issued in accordance with this version that have not yet expired.

E.2 Purpose

Compact encoding is applied when limited memory capacity is available for the IDL application (not exceeding 5 kB). This scheme requires the read device to read all data at the same time, after which the data is then parsed.

For compact encoding, a typical minimum capacity of 300 usable bytes is required. Typical media on which compact encoding is implemented are:

— 2D bar codes,

— RFID, and

— PICs and ICs with contacts with limited capacity available for the IDL application.

The limited storage capacity means that the number of data groups is restricted, as is the data size of each. The compact encoding scheme accordingly provides for Data Group 1, and optionally for any combination of data groups 2, 3, 4, 7 and 11, subject to storage capacity availability.

Means for authenticating the stored data are specified in ISO/IEC 18013-3.

E.3 Overview

The compact encoding method generates one data string containing all data groups. This data string is written to (and read from) storage media in the format provided for by each technology.

The compact encoding method differentiates between the following two types of data groups:

— Type 1 data group: Data groups that contain only data of which the allowable characters are specified in this document (i.e. data groups 1, 2 and 3).

— Type 2 data group: Data groups that include data of which the allowable characters are specified by another standard (i.e. data groups 4 and 7), and which thus may include delimiters as part of the field content.

Data Group 11 can be either a Type 1 or a Type 2 data group, depending on the information stored in this Data Group.

The encoding of data in a Type 1 data group is based on a fixed sequence of possible data elements in the data file. Each data field and data group (including optional and/or empty data fields and data groups) is terminated by an "end of field" or "end of data group" indicator. Data fields read sequentially from the data file thus can be assigned the appropriate data field name. This encoding method does not require each field to be identified individually with a tag in the data file, nor does it require the length of any field to be included in the data file.

The parsing rules for a Type 2 data group do not depend on delimiters to identify the data fields. Sufficient information is supplied in the data to calculate the position of the first and last bytes of each data field in the data stream.

E.4 Character set encoding

Unless otherwise specified, data objects are encoded as indicated in Table E.1.

Table E.1 — Encoding rules

Object	Encoding
Data fields of which the abstract values are defined as consisting of only N characters	BCD
Data fields of which the abstract values are defined as containing (although not necessarily exclusively) any A or S characters	ISO/IEC 8859-1 shall apply
Delimiters	ISO/IEC 8859-1 shall apply
Data object length	ASN.1
NOTE A field that is defined in this document as containing (amongst others) A and/or S characters will always be encoded using ISO/IEC 8859-1, even if an issuing authority's implementation of the same field is limited to N characters.	

E.5 Structure

E.5.1 Data file

The structure of a data file created using compact encoding can be represented as follows:

[header] × [Data Group 1] × [Data Group 2] × [Data Group 3] × [Data Group 4] × [Data Group 7] × [Data Group 11]¶

The header and Data Group 1 are mandatory; all other elements of the data file are optional. Data groups are delimited using the data group delimiter (×). The number of data group delimiters is fixed regardless of the number of optional data groups actually present. Data Group 11 is followed by the end of file delimiter (¶).

NOTE The data group delimiter (×) is a multiplication sign and not a lowercase X. Spaces (periods) have been inserted before and after the data group delimiter and before the end of file delimiter above for ease of reading only.

E.5.2 Header

The header consists of the following components:

[AID] [version] [length]

where

AID is the application identifier, 7 bytes; consisting of a 5 byte Registered Application Identifier (RID), and a 2 byte Proprietary Application Identifier Extension (PIX);

NOTE Separate PIXs are used for compact encoding (depending on the media used) and standard encoding — see Clause 9 for the values of the RID and the PIX;

Version is the 2 byte number; the first byte is assigned by ISO/IEC JTC 1/SC 17 for each new version of this document; The value of the byte for this version shall be '01'; the second byte is assigned by the issuing authority for each new version of their specification controlling the coding of domestic data (Data Group 11);

Length is the length of the data file (in bytes), encoded using ASN.1; the length equals the total number of bytes from (and including) the data group delimiter between the header and Data Group 1, up to and including the last character of the Logical Data Structure (LDS) (i.e. the end of file delimiter).

NOTE Although it is strictly speaking not necessary to know the length of the data file, it is included to assist in read verification.

E.5.3 Type 1 data group

A Type 1 data group consists of data elements delimited by the field delimiter (÷) as follows:

…× [element_1] ÷…÷ [element_n] ÷…÷ [element_last] ×…

All data elements are delimited (including optional elements), regardless of whether or not an element contains data. The only exception is if the data group contains no data, in which case no field delimiters are used. To facilitate forward compatibility, parsers shall be able to accommodate additional elements appended to a data group.

A data element can be sub-divided into data sub-fields. In a Type 1 data group, sub-fields are delimited by a sub-field delimiter [(;) sub-delimiter for short] as follows:

…[element_2] ÷ [field_3.1] ; [field_3.2] ; [field_3.3] ÷ [element_4]…

If a data sub-field is the last data element in a data group, it is terminated with the data group delimiter.

For data elements containing a fixed number of data sub-fields (e.g. the address field), the number of sub-delimiters is constant, regardless of the number of optional sub-fields present. The only exception is if none of the sub-fields contain data, in which case no sub-field delimiters are present.

The set of sub-fields in a data field may be repeated. If a set of sub-fields is not terminated with a field delimiter or a data group delimiter, it means that the next field will be the first sub-field of another set of sub-fields.

E.5.4 Type 2 data group

The contents of a Type 2 data group can generally be represented as follows:

× [fixed_length_field_1] [fixed_length_field_2] … [fixed_length_field_n] [variable_length_field_length] [variable_length_field] ×

where × is the data group delimiter. The length of a variable_length_field is specified using ASN.1 rules (see E.6). The number of fixed length fields and the number of variable length fields is not restricted. The number and sequence of fields are specified in the data group definition.

E.6 Implementation

E.6.1 Data Group 1: mandatory data

Data Group 1 is a Type 1 data group.

A sub-field delimiter is used between different instances of the category of vehicle/restriction/condition data object.

E.6.2 Data Group 2: optional licence holder information

Data Group 2 is a Type 1 data group.

E.6.3 Data Group 3: optional issuing authority details

Data Group 3 is a Type 1 data group.

The document discriminator field as well as the data discriminator field shall be each encoded as a 1 byte binary number. The ISO issuer ID number field shall be encoded as a 4 byte BCD number.

E.6.4 Data Group 4: optional portrait images

For compact encoding, Data Group 4 supports one portrait image only. Consequently, not all of the fields defined in 8.5 are provided for. The coding of the portrait image is specified outside of this document, and thus Data Group 4 is a Type 2 data group. Data Group 4 is coded as follows (spaces are included to enhance legibility only, and are not encoded):

[previous data group] × [type of image] [image length] [image] × [next data group]

where

[type of image]	is a fixed length field;
[image length]	is the length of the [image] field, expressed using ASN.1 rules;
[image]	is a variable length field;
the [image] field	is encoded as a binary object.

E.6.5 Data Group 5: optional signature/mark image

Data Group 5 is not supported in compact encoding.

E.6.6 Data Group 6: optional facial biometric template

Data Group 6 is not supported in compact encoding.

E.6.7 Data Group 7: optional finger template

Data Group 7 is a Type 2 data group. Due to limited storage space, only finger minutiae data and finger pattern spectral data are supported in Data Group 7. This limitation precludes the use of optional data elements listed in Table 7.

Data Group 7 thus is coded as follows (spaces are included to enhance legibility only, and are not encoded):

[previous data group] × [BDB format owner] [BDB format type] [biometric data block length] [biometric data block] × [next data group]

where

[BDB format owner]	is a fixed length field;
[BDB format type]	is a fixed length field;
[biometric data block length]	is the length of the [biometric data block] field, expressed using ASN.1 rules;
[biometric data block]	is a variable length field, encoded in accordance with Table B.1, with the understanding that delimiters may be included in data fields as data (i.e. without fulfilling a delimiting function).