[F1[F2ANNEX I B

REQUIREMENTS FOR CONSTRUCTION, TESTING, INSTALLATION AND INSPECTION

Textual Amendments

F1 Inserted by Council Regulation (EC) No 2135/98 of 24 September 1998 amending Regulation (EEC) No 3821/85 on recording equipment in road transport and Directive 88/599/EEC concerning the application of Regulations (EEC) No 3820/85 and (EEC) No 3821/85.

F2 Substituted by Commission Regulation (EC) No 1360/2002 of 13 June 2002 adapting for the seventh time to technical progress Council Regulation (EEC) No 3821/85 on recording equipment in road transport (Text with EEA relevance).

Appendix 2

TACHOGRAPH CARDS SPECIFICATION

3. HARDWARE AND COMMUNICATION

3.1. Introduction

This paragraph describes the minimum functionality required by Tachograph cards and VUs to ensure correct operation and interoperability.

Tachograph cards are as compliant as possible with the available ISO/IEC applicable norms (especially ISO/IEC 7816). However, commands and protocols are fully described in order to specify some restricted usage or some differences if they exist. The commands specified are fully compliant with the referred norms except where indicated.

3.2. Transmission protocol

The Transmission protocol shall be compliant with ISO/IEC 7816-3. In particular, the VU shall recognise waiting time extensions sent by the card.

3.2.1. Protocols

The card shall provide both protocol T=0 and protocol T=1.

T=0 is the default protocol, a PTS command is therefore necessary to change the protocol to T=1.

Devices shall support direct convention in both protocols: the direct convention is hence mandatory for the card.

The Information Field Size Card byte shall be presented at the ATR in character TA3. This value shall be at least ′F0h′ (= 240 bytes).

The following restrictions apply to the protocols:

T=0

— The interface device shall support an answer on I/O after the rising edge of the signal on RST from 400 cc.

— The interface device shall be able to read characters separated with 12 etu.

— The interface device shall read an erroneous character and its repetition if separated with 13 etu. If an erroneous character is detected, the Error signal on I/O can occur between 1 etu and 2 etu. The device shall support a 1 etu delay.

— The interface device shall accept a 33 bytes ATR (TS+32)

— If TC1 is present in the ATR, the Extra Guard Time shall be present for characters sent by the interface device although characters sent by the card can still be separated with 12 etu. This is also true for the ACK character sent by the card after a P3 character emitted by the interface device.

— The interface device shall take into account a NUL character emitted by the card.

— The interface device shall accept the complementary mode for ACK.

— The get-response command cannot be used in chaining mode to get a data which length could exceed 255 bytes.

T=1

— NAD byte: not used (NAD shall be set to ′00′).
— S-block ABORT: not used.
— S-block VPP state error: not used.
— The total chaining length for a data field will not exceed 255 bytes (to be ensured by the IFD).
— The Information Field Size Device (IFSD) shall be indicated by the IFD immediately after the ATR: the IFD shall transmit the S-Block IFS request after the ATR and the card shall send back S-Block IFS. The recommended value for IFSD is 254 bytes.
— The card will not ask for an IFS readjustment.

3.2.2. ATR

The device checks ATR bytes, according to ISO/IEC 7816-3. No verification shall be done on ATR Historical Characters.

EXAMPLE OF BASIC BIPROTOCOL ATR ACCORDING TO ISO/IEC 7816-3

Character	Value	Remarks
TS	′3Bh′	Indicates direct convention
T0	′85h′	TD1 present; 5 historical bytes are presents
TD1	′80h′	TD2 present; T=0 to be used
TD2	′11h′	TA3 present; T=1 to be used
TA3	′XXh′ ([X1at least] ′F0h′)	Information Field Size Card (IFSC)
TH1 bis TH5	′XXh′	Historical characters
TCK	′XXh′	Check character (exclusive OR)

Editorial Information

X1 Substituted by Corrigendum to Commission Regulation (EC) No 1360/2002 of 13 June 2002 adapting for the seventh time to technical progress Council Regulation (EEC) No 3821/85 on recording equipment in road transport (Official Journal of the European Communities L 207 of 5 August 2002).

After the Answer To Reset (ATR), the Master File (MF) is implicitly selected and becomes the Current Directory.

3.2.3. PTS

The default Protocol is T=0. To set the T=1 protocol, a PTS (also known as PPS) must be sent to the card by the device.

As both T=0 and T=1 protocols are mandatory for the card, the basic PTS for protocol switching is mandatory for the card.

The PTS can be used, as indicated in ISO/IEC 7816-3, to switch to higher baud rates than the default one proposed by the card in the ATR if any (TA(1) byte).

Higher baud rates are optional for the card.

If no other baud rate than the default one are supported (or if the selected baud rate is not supported), the card shall respond to the PTS correctly according to ISO/IEC 7816-3 by omitting the PPS1 byte.

Examples of basic PTS for protocol selection are the following:

Character	Value	Remarks
PPSS	′FFh′	The initiate character
PPS0	′00h′ or ′01h′	PPS1 to PPS3 are not present; ′00h′ to select T0, ′01h′ to select T1
PK	′XXh′	Check character: ′XXh′ = ′FFh′ if PPS0 = ′00h′ ′XXh′ = ′FEh′ if PPS0 = ′01h′

3.3. Access conditions (AC)

Access Conditions (AC) for the UPDATE_BINARY and READ_BINARY commands are defined for each Elementary File.

The AC of the current file must be met before accessing the file via these commands.

The definitions of the available access conditions are the following:

—ALW: The action is always possible and can be executed without any restriction.
—NEV: The action is never possible.
—AUT: The right corresponding a successful external authentication must be opened up (done by the EXTERNAL_AUTHENTICATE command).
—PRO SM: Command must be transmitted with a cryptographic checksum using secure messaging (See Appendix 11).
—AUT and PRO SM (combined)

On the processing commands (UPDATE_BINARY and READ_BINARY), the following access conditions can be set in the card:

	UPDATE_ BINARY	READ_ BINARY
ALW	Yes	Yes
NEV	Yes	Yes
AUT	Yes	Yes
PRO SM	Yes	No
AUT and PRO SM	Yes	No

The PRO SM access condition is not available for the READ_BINARY command. It means that the presence of a cryptographic checksum for a READ command is never mandatory. However, using the value ′OC′ for the class, it is possible to use the READ_BINARY command with secure messaging, as described in paragraph 3.6.2.

3.4. Data encryption

When confidentiality of data to be read from a file needs to be protected, the file is marked as 'Encrypted'. Encryption is performed using secure messaging (See Appendix 11).

3.5. Commands and error codes overview

Commands and file organisation are deduced from and complies with ISO/IEC 7816-4.

This section describes the following APDU command-response pairs:

Command	**INS**
SELECT FILE	A4
READ BINARY	B0
UPDATE BINARY	D6
GET CHALLENGE	84
VERIFY	20
GET RESPONSE	C0
PERFORM SECURITY OPERATION: VERIFY CERTIFICATE COMPUTE DIGITAL SIGNATURE VERIFY DIGITAL SIGNATURE HASH	2A
INTERNAL AUTHENTICATE	88
EXTERNAL AUTHENTICATE	82
MANAGE SECURITY ENVIRONMENT: SETTING A KEY	22
PERFORM HASH OF FILE	2A

The status word SW1 SW2 are returned in any response message and denote the processing state of the command.

SW1	**SW2**	**Meaning**
90	00	Normal processing
61	XX	Normal processing. XX = number of response bytes available
62	81	Warning processing. Part of returned data may be corrupted
63	CX	Wrong CHV (PIN). Remaining attempts counter provided by 'X'

64	00	Execution error — State of non-volatile memory unchanged. Integrity error
65	00	Execution error — State of non-volatile memory changed
65	81	Execution error — State of non-volatile memory changed — Memory failure
66	88	Security error — wrong Cryptographic checksum (during secure messaging) or wrong certificate (during certificate verification) or wrong cryptogram (during external authentication) or wrong signature (during signature verification)
67	00	Wrong length (wrong Lc or Le)
69	00	Forbidden command (no response available in T=0)
69	82	Security status not satisfied
69	83	Authentication method blocked
69	85	Conditions of use not satisfied
69	86	Command not allowed (no current EF)
69	87	Expected secure messaging data objects missing
69	88	Incorrect secure messaging data objects
6A	82	File not found
6A	86	Wrong parameters P1-P2
6A	88	Referenced data not found
6B	00	Wrong parameters (offset outside the EF)
6C	XX	Wrong length, SW2 indicates the exact length. No data field is returned

6D	00	Instruction code not supported or invalid
6E	00	Class not supported
6F	00	Other checking errors

3.6. Commands description

The mandatory commands for the Tachograph cards are described in this chapter.

Additional relevant details, related to cryptographic operations involved, are given in Appendix 11 Common security mechanisms.

All commands are described independently of the used protocol (T=0 or T=1). The APDU bytes CLA, INS, P1, P2, Lc and Le are always indicated. If Lc or Le is not needed for the described command, the associated length, value and description are empty.

If both length bytes (Lc and Le) are requested, the described command has to be split in two parts if the IFD is using protocol T=0: the IFD sends the command as described with P3=Lc + data and then sends a GET_RESPONSE (see point 3.6.6) command with P3=Le.

If both length bytes are requested, and Le=0 (secure messaging):

— When using protocol T=1, the card shall answer to Le=0 by sending all available output data.

— When using protocol T=0, the IFD shall send the first command with P3=Lc + data, the card shall answer (to this implicit Le=0) by the Status bytes ′61La′, where La is the number of response bytes available. The IFD shall then generate a GET RESPONSE command with P3=La to read the data.

3.6.1. Select file

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The SELECT FILE command is used:

— to select an application DF (selection by name must be used)

— to select an elementary file corresponding to the submitted file ID

3.6.1.1. Selection by name (AID)

This command allows to select an application DF in the card.

This command can be performed from anywhere in the file structure (after the ATR or at anytime).

The selection of an application resets the current security environment. After performing the application selection, no current public key is selected anymore and the former session key is no longer available for secure messaging. The AUT access condition is also lost.

COMMAND MESSAGE

Byte	**Length**	**Value**	**Description**
CLA	1	′00h′	
INS	1	′A4h′	

P1	1	′04h′	Selection by name (AID)
P2	1	′0Ch′	No response expected
Lc	1	′NNh′	Number of bytes sent to the card (length of the AID): ′06h′ for the Tachograph application
#6-#(5+NN)	NN	′XX..XXh′	AID: ′FF 54 41 43 48 4F′ for the Tachograph application

No response to the SELECT FILE command is needed (Le absent in T=1, or no response asked in T=0).

RESPONSE MESSAGE (NO RESPONSE ASKED)

Byte	Length	Value	Description
SW	2	′XXXXh′	Status Words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if the application corresponding with the AID is not found, the processing state returned is ′6A82′,
— in T=1, if the byte Le is present, the state returned is ′6700′,
— in T=0, if a response is asked after the SELECT FILE command, the state returned is ′6900′,
— if the selected application is considered corrupted (integrity error is detected within the file attributes), the processing state returned is ′6400′ or ′6581′.

3.6.1.2. Selection of an elementary file using its file identifier

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	
INS	1	′A4h′	
P1	1	′02h′	Selection of an EF under the current DF
P2	1	′0Ch′	No response expected
Lc	1	′02h′	Number of bytes sent to the card
#6-#7	2	′XXXXh′	File Identifier

No response to the SELECT FILE command is needed (Le absent in T=1, or no response asked in T=0).

RESPONSE MESSAGE (NO RESPONSE ASKED)

Byte	Length	Value	Description
SW	2	′XXXXh′	Status Words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if the file corresponding with the file identifier is not found, the processing state returned is ′6A82′,
— in T=1, if the byte Le is present, the state returned is ′6700′,
— in T=0, if a response is asked after the SELECT FILE command, the state returned is ′6900′,
— if the selected file is considered corrupted (integrity error is detected within the file attributes), the processing state returned is ′6400′ or ′6581′.

3.6.2. Read Binary

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The Read Binary command is used to read data from a transparent file.

The response of the card consists of returning the data read, optionally encapsulated in a secure messaging structure.

The command can be performed only if the security status satisfies the security attributes defined for the EF for the READ function.

3.6.2.1. Command without secure messaging

This command enables the IFD to read data from the EF currently selected, without secure messaging.

Reading data from a file marked as ‘Encrypted’ shall not be possible through this command.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	No secure messaging asked
INS	1	′B0h′	
P1	1	′XXh′	Offset in bytes from the beginning of the file: most significant byte
P2	1	′XXh′	Offset in bytes from the beginning of the file: least significant byte

Le	1	′XXh′	Length of data expected. number of bytes to be read

Note: bit 8 of P1 must be set to 0.

RESPONSE MESSAGE

Byte	Length	Value	Description
#1-#X	X	′XX..XXh′	Data read
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if no EF is selected, the processing state returned is ′6986′,
— if the Access Control of the selected file are not satisfied, the command is interrupted with ′6982′,
— if the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is ′6B00′,
— if the size of the data to be read is not compatible with the size of the EF (Offset + Le > EF size) the processing state returned is ′6700′ or ′6Cxx′, where ′xx′ indicates the exact length,
— if an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is ′6400′ or ′6581′,
— if an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is ′6281′.

3.6.2.2. Command with secure messaging

This command enables the IDF to read data from the EF currently selected with secure messaging, in order to verify the integrity of the data received and to protect the confidentiality of the data in the case the EF is marked as ‘Encrypted’.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′0Ch′	Secure Messaging asked
INS	1	′B0h′	INS
P1	1	′XXh′	P1 (offset in bytes from the beginning of the file): Most Significant Byte
P2	1	′XXh′	P2 (offset in bytes from the beginning of the file): Least Significant Byte

Lc	1	′09h′	Length of input data for secure messaging
#6	1	′97h′	T_{LE}: Tag for expected length specification
#7	1	′01h′	L_{LE}: Length of expected length
#8	1	′NNh′	Expected length specification (original Le): Number of Bytes to be read
#9	1	′8Eh′	T_{CC}: Tag for cryptographic checksum
#10	1	′04h′	L_{CC}: Length of following cryptographic checksum
#11-#14	4	′XX..XXh′	Cryptographic checksum (4 most significant bytes)
Le	1	′00h′	As specified in ISO/ IEC 7816-4

Response Message if EF is not marked as ‘Encrypted’ and if Secure Messaging input format is correct:

Byte	**Length**	**Value**	**Description**
#1	1	′81h′	T_{PV}: Tag for plain value data
#2	L	′NNh′ or ′81 NNh′	L_{PV}: length of returned data (= original Le) L is 2 bytes if L_{PV} >127 bytes
#(2+L)-#(1+L+NN)	NN	′XX..XXh′	Plain Data value
#(2+L+NN)	1	′8Eh′	T_{CC}: Tag for cryptographic checksum
#(3+L+NN)	1	′04h′	L_{CC}: Length of following cryptographic checksum

#(4+L+NN)-#(7+L+NN)	4	′XX..XXh′	Cryptographic checksum (4 most significant bytes)
SW	2	′XXXXh′	Status Words (SW1, SW2)

Response Message if EF is marked as ‘Encrypted’ and if Secure Messaging input format is correct:

Byte	Length	Value	Description
#1	1	′87h′	$T_{PI\ CG}$: Tag for encrypted data (cryptogram)
#2	L	′MMh′ or ′81 MMh′	$L_{PI\ CG}$: length of returned encrypted data (different of original Le of the command due to padding) L is 2 byte if $L_{PI\ CG}$> 127 bytes
#(2+L)-#(1+L+MM)	MM	′01XX..XXh′	Encrypted data: padding indicator and cryptogram
#(2+L+MM)	1	′8Eh′	T_{CC}: tag for cryptographic checksum
#(3+L+MM)	1	′04h′	L_{CC}: length of following cryptographic checksum
#(4+L+MM)-#(7+L+MM)	4	′XX..XXh′	Cryptographic checksum (4 most significant bytes)
SW	2	′XXXXh′	Status words (SW1, SW2)

The encrypted data returned contain a first byte indicating the used padding mode. For the tachograph application, the padding indicator always takes the value '01h', indicating that the used padding mode is the one specified in ISO/IEC 7816-4 (one byte with value ′80h′ followed by some null bytes: ISO/IEC 9797 method 2).

The ‘regular’ processing states, described for the READ BINARY command with no secure messaging (see point 3.6.2.1), can be returned using the response message structures described above, under a ′99h′ Tag (as described in TCS 335).

Additionally, some errors specifically related to secure messaging can happen. In that case, the processing state is simply returned, with no secure messaging structure involved:

RESPONSE MESSAGE IF INCORRECT SECURE MESSAGING INPUT FORMAT

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (SW1, SW2)

— If no current session key is available, the processing state ′6A88′ is returned. It happens either if the session key has not already been generated or if the session key validity has expired (in this case the IFD must re-run a mutual authentication process to set a new session key).

— If some expected data objects (as specified above) are missing in the secure messaging format, the processing state ′6987′ is returned: this error happens if an expected tag is missing or if the command body is not properly constructed.

— If some data objects are incorrect, the processing state returned is ′6988′: this error happens if all the required tags are present but some lengths are different from the ones expected.

— If the verification of the cryptographic checksum fails, the processing state returned is ′6688′.

3.6.3. Update Binary

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The UPDATE BINARY command message initiates the update (erase + write) of the bits already present in an EF binary with the bits given in the command APDU.

The command can be performed only if the security status satisfies the security attributes defined for the EF for the UPDATE function (If the access control of the UPDATE function includes PRO SM, a secure messaging must be added in the command).

3.6.3.1. Command without secure messaging

This command enables the IFD to write data into the EF currently selected, without the card verifying the integrity of data received. This plain mode is allowed only if the related file is not marked as ‘Encrypted’.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	No secure messaging asked
INS	1	′D6h′	
P1	1	′XXh′	Offset in bytes from the beginning of the file: most significant byte
P2	1	′XXh′	Offset in bytes from the beginning of the file: least significant byte

Lc	1	'NNh'	Lc length of data to Update. Number of bytes to be written
#6-#(5+NN)	NN	'XX..XXh'	Data to be written

Note: bit 8 of P1 must be set to 0.

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	'XXXXh'	Status words (SW1, SW2)

— If the command is successful, the card returns '9000',
— if no EF is selected, the processing state returned is '6986',
— if the Access Control of the selected file are not satisfied, the command is interrupted with '6982',
— if the Offset is not compatible with the size of the EF (Offset > EF size), the processing state returned is '6B00',
— if the size of the data to be written is not compatible with the size of the EF **[**[F3](Offset + Lc > EF size)**]** the processing state returned is '6700',
— if an integrity error is detected within the file attributes, the card shall consider the file as corrupted and unrecoverable, the processing state returned is '6400' or '6500',
— if writing is unsuccessful, the processing state returned is '6581'.

Textual Amendments

F3 Substituted by Commission Regulation (EC) No 432/2004 of 5 March 2004 adapting for the eighth time to technical progress Council Regulation (EEC) No 3821/85 of 20 December 1985 on recording equipment in road transport (Text with EEA relevance).

3.6.3.2. Command with secure messaging

This command enables the IFD to write data into the EF currently selected, with the card verifying the integrity of data received. As no confidentiality is required, the data are not encrypted.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure messaging. Asked
INS	1	'D6h'	INS
P1	1	'XXh'	Offset in bytes from the beginning of the file: most significant byte

P2	1	′XXh′	Offset in bytes from the beginning of the file: least significant byte
Lc	1	′XXh′	Length of the secured data field
#6	1	′81h′	T_{PV}: tag for plain value data
#7	L	′NNh′ or ′81 NNh′	L_{PV}: length of transmitted data L is 2 bytes if L_{PV}> 127 bytes
#(7+L)-#(6+L+NN)	NN	′XX..XXh′	Plain data value (data to be written)
#(7+L+NN)	1	′8Eh′	T_{CC}: tag for cryptographic checksum
#(8+L+NN)	1	′04h′	L_{CC}: Length of following cryptographic checksum
#(9+L+NN)-#(12+L+NN)	4	′XX..XXh′	Cryptographic checksum (4 most significant bytes)
Le	1	′00h′	As specified in ISO/IEC 7816-4

RESPONSE MESSAGE IF CORRECT SECURE MESSAGING INPUT FORMAT

Byte	**Length**	**Value**	**Description**
#1	1	′99h′	T_{SW}: tag for status words (to be protected by CC)
#2	1	′02h′	L_{SW}: length of returned status words
#3-#4	2	′XXXXh′	Status words (SW1, SW2)
#5	1	′8Eh′	T_{CC}: tag for cryptographic checksum
#6	1	′04h′	L_{CC}: Length of following cryptographic checksum

#7-#10	4	′XX..XXh′	Cryptographic checksum (4 most significant bytes)
SW	2	′XXXXh′	Status words (SW1, SW2)

The ‘regular’ processing states, described for the UPDATE BINARY command with no secure messaging (see point 3.6.3.1), can be returned using the response message structure described above.

Additionally, some errors specifically related to secure messaging can happen. In that case, the processing state is simply returned, with no secure messaging structure involved:

RESPONSE MESSAGE IF ERROR IN SECURE MESSAGING

Byte	Length	Value	Description
SW	2	′XXXXh′	Status Words (SW1, SW2)

— If no current session key is available, the processing state ′6A88′ is returned,

— if some expected data objects (as specified above) are missing in the secure messaging format, the processing state ′6987′ is returned: this error happens if an expected tag is missing or if the command body is not properly constructed,

— if some data objects are incorrect, the processing state returned is ′6988′: this error happens if all the required tags are present but some lengths are different from the ones expected,

— if the verification of the cryptographic checksum fails, the processing state returned is ′6688′.

3.6.4. Get challenge

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The GET CHALLENGE command asks the card to issue a challenge in order to use it in a security related procedure in which a cryptogram or some ciphered data are sent to the card.

The Challenge issued by the card is only valid for the next command, which uses a challenge, sent to the card.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′84h′	INS
P1	1	′00h′	P1
P2	1	′00h′	P2
Le	1	′08h′	Le (Length of challenge expected)

RESPONSE MESSAGE

Byte	Length	Value	Description
#1-#8	8	′XX..XXh′	Challenge
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if Le is different from ′08h′, the processing state is ′6700′,
— if parameters P1-P2 are incorrect, the processing state is ′6A86′.

3.6.5. Verify

This command is compliant with ISO/IEC 7816-4, but has a restricted usage compared to the command defined in the norm.

The Verify command initiates the comparison in the card of the CHV (PIN) data sent from the command with the reference CHV stored in the card.

Note: The PIN entered by the user must be right padded with FFh' bytes up to a length of 8 bytes by the IFD.

If the command is successful, the rights corresponding to CHV presentation are opened and the remaining CHV attempt counter is reinitialised.

An unsuccessful comparison is recorded in the card in order to limit the number of further attempts of the use of the reference CHV.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′20h′	INS
P1	1	′00h′	P1
P2	1	′00h′	P2 (the verified CHV is implicitly known)
Lc	1	′08h′	Length of CHV code transmitted
#6-#13	8	′XX..XXh′	CHV

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if the reference CHV is not found, the processing state returned is ′6A88′,

— if the CHV is blocked, (the remaining attempt counter of the CHV is null), the processing state returned is ′6983′. Once in that state, the CHV can never be successfully presented anymore,
— if the comparison is unsuccessful, the remaining attempt Counter is decreased and the status ′63CX′ is returned (X > 0, and X equals the remaining CHV attempts counter. X = 'F′, the CHV attempts counter is greater than 'F′),
— if the reference CHV is considered corrupted, the processing state returned is ′6400′ or ′6581′.

3.6.6. Get response

This command is compliant with ISO/IEC 7816-4.

This command (only necessary and available for T=0 Protocol) is used to transmit prepared data from the card to the interface device (case where a command had included both Lc and Le).

The GET_RESPONSE command has to be issued immediately after the command preparing the data, otherwise, the data are lost. After the execution of the GET_RESPONSE command (except if the error ′61xx′ or ′6Cxx′ occur, see below), the previously prepared data are no longer available.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	
INS	1	′C0h′	
P1	1	′00h′	
P2	1	′00h′	
Le	1	′XXh′	Number of bytes expected

RESPONSE MESSAGE

Byte	Length	Value	Description
#1-#X	X	′XX..XXh′	Data
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′.
— If no data have been prepared by the card, the processing state returned is ′6900′ or ′6F00′.
— If Le exceeds the number of available bytes or if Le is null, the processing state returned is ′6Cxx′, where ′xx′ denotes the exact number of available bytes. In that case, the prepared data are still available for a subsequent GET_RESPONSE command.
— If Le is not null and is smaller than the number of available bytes, the required data are sent normally by the card, and the processing state returned is ′61xx′, where ′xx ′ indicates a number of extra bytes still available by a subsequent GET_RESPONSE command.
— If the command is not supported (protocol T=1), the card returns ′6D00′.

3.6.7. PSO: verify certificate

This command is compliant with ISO/IEC 7816-8, but has a restricted usage compared to the command defined in the norm.

The VERIFY CERTIFICATE command is used by the card to obtain a Public Key from the outside and to check its validity.

When a VERIFY CERTIFICATE command is successful, the Public Key is stored for a future use in the Security environment. This key shall be explicitly set for the use in security related commands (INTERNAL AUTHENTICATE, EXTERNAL AUTHENTICATE or VERIFY CERTIFICATE) by the MSE command (see point 3.6.10) using its key identifier.

In any case, the VERIFY CERTIFICATE command uses the public key previously selected by the MSE command to open the certificate. This public key must be the one of a Member State or of Europe.

COMMAND MESSAGE

Byte	**Length**	**Value**	**Description**
CLA	1	′00h′	CLA
INS	1	′2Ah′	Perform Security Operation
P1	1	′00h′	P1
P2	1	′AEh′	P2: non BER-TLV coded data (concatenation of data elements)
Lc	1	′⌊F3⌋C2h⌋′	Lc: Length of the certificate, 194 bytes
#6-#199	194	′XX..XXh′	Certificate: concatenation of data elements (as described in Appendix 11)

RESPONSE MESSAGE

Byte	**Length**	**Value**	**Description**
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if the certificate verification fails, the processing state returned is ′6688′. The verification and unwrapping process of the certificate is described in Appendix 11,

— if no Public Key is present in the Security Environment, ′6A88′ is returned,

— if the selected public key (used to unwrap the certificate) is considered corrupted, the processing state returned is ′6400′ or ′6581′,

— if the selected public key (used to unwrap the certificate) has a CHA.LSB (*CertificateHolderAuthorisation.equipmentType*) different from ′00′ (i.e. is not the one of a Member State or of Europe), the processing state returned is ′6985′.

3.6.8. Internal authenticate

This command is compliant with ISO/IEC 7816-4.

Using the INTERNAL AUTHENTICATE command, the IFD can authenticate the card.

The authentication process is described in Appendix 11. It includes the following statements:

The INTERNAL AUTHENTICATE command uses the card Private Key (implicitly selected) to sign authentication data including K1 (first element for session key agreement) and RND1, and uses the Public Key currently selected (through the last MSE command) to encrypt the signature and form the authentication token (more details in Appendix 11).

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′88h′	INS
P1	1	′00h′	P1
P2	1	′00h′	P2
Lc	1	′10h′	Length of data sent to the card
#6-#13	8	′XX..XXh′	Challenge used to authenticate the card
#14-#21	8	′XX..XXh′	VU.CHR (see Appendix 11)
Le	1	′80h′	Length of the data expected from the card

RESPONSE MESSAGE

Byte	Length	Value	Description
#1-#128	128	′XX..XXh′	Card authentication token (see Appendix 11)
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,
— if no Public Key is present in the Security Environment, the processing state returned is ′6A88′,
— if no Private Key is present in the Security Environment, the processing state returned is ′6A88′,

— if VU.CHR does not match the current public key identifier, the processing state returned is ′6A88′,
— if the selected private key is considered corrupted, the processing state returned is ′6400′ or ′6581′.

If the INTERNAL_AUTHENTICATE command is successful, the current session key, if existing, is erased and no longer available. In order to have a new session key available, the EXTERNAL_AUTHENTICATE command must be successfully performed.

3.6.9. External authenticate

This command is compliant with ISO/IEC 7816-4.

Using the EXTERNAL AUTHENTICATE command, the card can authenticate the IFD.

The authentication process is described in Appendix 11. It includes the following statements:

A GET CHALLENGE command must precede the EXTERNAL_AUTHENTICATE command immediately. The card issues a challenge to the outside (RND3).

The verification of the cryptogram uses RND3 (challenge issued by the card), the card private key (implicitly selected) and the public key previously selected by the MSE command.

The card verifies the cryptogram, and if it is correct, the AUT access condition is opened.

[[X1]The input cryptogram carries the second element for session key agreement K2.**]**

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′82h′	INS
P1	1	′00h′	P1
P2	1	′00h′	P2 (the public key to be used is implicitly known, and has been previously set by the MSE command)
Lc	1	′80h′	Lc (Length of the data sent to the card)
#6-#133	128	′XX..XXh′	Cryptogram (see Appendix 11)

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (status words (SW1, SW2))

— If the command is successful, the card returns ′9000′,
— if no Public Key is present in the Security Environment, ′6A88′ is returned,

— if the CHA of the currently set public key is not the concatenation of the Tachograph application AID and of a VU equipment Type, the processing state returned is ′6F00′ (see Appendix 11),
— if no Private Key is present in the Security Environment, the processing state returned is ′6A88′,
— if the verification of the cryptogram is wrong, the processing state returned is ′6688′,
— if the command is not immediately preceded with a GET CHALLENGE command, the processing state returned is ′6985′,
— if the selected private key is considered corrupted, the processing state returned is ′6400′ or ′6581′.

If the EXTERNAL AUTHENTICATE command is successful, and if the first part of the session key is available from a successful INTERNAL AUTHENTICATE recently performed, the session key is set for future commands using secure messaging.

If the first session key part is not available from a previous INTERNAL AUTHENTICATE command, the second part of the session key, sent by the IFD, is not stored in the card. This mechanism ensures that the mutual authentication process is done in the order specified in Appendix 11.

3.6.10. Manage security environment

This command is used to set a public key for authentication purpose.

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

The key referenced in the MSE data field is valid for every file of the Tachograph DF.

The key referenced in the MSE data field remains the current public key until the next correct MSE command.

If the key referenced is not (already) present into the card, the security environment remains unchanged.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′22h′	INS
P1	1	′C1h′	P1: referenced key valid for all cryptographic operations
P2	1	′B6h′	P2 (referenced data concerning digital signature)
Lc	1	′0Ah′	Lc: length of subsequent data field

#6	1	′83h′	Tag for referencing a public key in asymmetric cases
#7	1	′08h′	Length of the key reference (key identifier)
#8-#15	08h	′XX..XXh′	Key identifier as specified in Appendix 11

RESPONSE MESSAGE

Byte	**Length**	**Value**	**Description**
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if the referenced key is not present into the card, the processing state returned is ′6A88′,

— if some expected data objects are missing in the secure messaging format, the processing state ′6987′ is returned. This can happen if the tag ′83h′ is missing,

— if some data objects are incorrect, the processing state returned is ′6988′. This can happen if the length of the key identifier is not ′08h′,

— if the selected key is considered corrupted, the processing state returned is ′6400′ or ′6581′.

3.6.11. PSO: hash

This command is used to transfer to the card the result of a hash calculation on some data. This command is used for the verification of digital signatures. The hash value is stored in EEPROM for the subsequent command verify digital signature.

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

COMMAND MESSAGE

Byte	**Length**	**Value**	**Description**
CLA	1	′00h′	CLA
INS	1	′2Ah′	Perform security operation
P1	1	′90h′	Return hash code
P2	1	′A0h′	Tag: data field contains DOs relevant for hashing
Lc	1	′16h′	Length Lc of the subsequent data field
#6	1	′90h′	Tag for the hash code

#7	1	′14h′	Length of the hash code
#8-#27	20	′XX..XXh′	Hash code

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if some expected data objects (as specified above) are missing, the processing state ′6987′ is returned. This can happen if one of the tag ′90h′ is missing,

— if some data objects are incorrect, the processing state returned is ′6988′. This error happens if the required tag is present but with a length different from ′14h′.

3.6.12. Perform hash of file

This command is not compliant with ISO/IEC 7816-8. Thus the CLA byte of this command indicates that there is a proprietary use of the PERFORM SECURITY OPERATION/HASH.

The perform hash file command is used to hash the data area of the currently selected transparent EF.

The result of the hash operation is stored in the card. It can then be used to get a digital signature of the file, using the PSO-COMPUTE_DIGITAL_SIGNATURE command. This result remains available for the COMPUTE DIGITAL SIGNATURE command until the next successful PERFORM HASH of FILE command.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′80h′	CLA
INS	1	′2Ah′	Perform security operation
P1	1	′90h′	Tag: hash
P2	1	′00h′	P2: hash the data of the currently selected transparent file

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if no application is selected, the processing state ′6985′ is returned,

— if the selected EF is considered corrupted (file attributes or stored data integrity errors), the processing state returned is ′6400′ or ′6581′,

— if the selected file is not a transparent file, the processing state returned is ′6986′.

3.6.13. PSO: compute digital signature

This command is used to compute the digital signature of previously computed hash code (see PERFORM HASH of FILE, point 3.6.12).

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

The card private key is used to compute the digital signature and is implicitly known by the card.

The card performs a digital signature using a padding method compliant with PKCS1 (see Appendix 11 for details).

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′2Ah′	Perform security operation
P1	1	′9Eh′	Digital signature to be returned
P2	1	′9Ah′	Tag: data field contains data to be signed. As no data field is included, the data are supposed to be already present in the card (hash of file)
Le	1	′80h′	Length of the expected signature

RESPONSE MESSAGE

Byte	Length	Value	Description
#1-#128	128	′XX..XXh′	Signature of the previously computed hash
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if the implicitly selected private key is considered as corrupted, the processing state returned is ′6400′ or ′6581′.

3.6.14. PSO: verify digital signature

This command is used to verify the digital signature, provided as an input, in accordance with PKCS1 of a message, whose hash is known to the card. The signature algorithm is implicitly known by the card.

This command is compliant with ISO/IEC 7816-8. The use of this command is restricted regarding the related standard.

The Verify Digital Signature command always uses the public key selected by the previous Manage Security Environment command, and the previous hash code entered by a PSO: hash command.

COMMAND MESSAGE

Byte	Length	Value	Description
CLA	1	′00h′	CLA
INS	1	′2Ah′	Perform security operation
P1	1	′00h′	
P2	1	′A8h′	Tag: data field contains DOs relevant for verification
Lc	1	′83h′	Length Lc of the subsequent data field
#28	1	′9Eh′	Tag for digital signature
#29-#30	2	′8180h′	Length of digital signature (128 bytes, coded in accordance with ISO/IEC 7816-6)
#31-#158	128	′XX..XXh′	Digital signature content

RESPONSE MESSAGE

Byte	Length	Value	Description
SW	2	′XXXXh′	Status words (SW1, SW2)

— If the command is successful, the card returns ′9000′,

— if the verification of the signature fails, the processing state returned is ′6688′. The verification process is described in Appendix 11,

— if no public key is selected, the processing state returned is ′6A88′,

— if some expected data objects (as specified above) are missing, the processing state ′6987′ is returned. This can happen if one of the required tag is missing,

— if no hash code is available to process the command (as a result of a previous PSO: hash command), the processing state returned is ′6985′,

— if some data objects are incorrect, the processing state returned is ′6988′. This can happen if one of the required data objects length is incorrect,

— if the selected public key is considered corrupted, the processing state returned is ′6400′ or ′6581′.**]]**

Status:
Point in time view as at 11/04/2007.

Changes to legislation:
There are currently no known outstanding effects for the Council Regulation (EEC) No 3821/85, Division 3..