



Economic and Social Council

Distr.: General
13 April 2021

Original: English

Economic Commission for Europe

Inland Transport Committee

Working Party on Road Transport

Group of Experts on European Agreement Concerning Work of Crews of Vehicles Engaged in International Road Transport (AETR)

Twenty-sixth session

Geneva, 14 June 2021

Item 2 (b) of the provisional agenda

Programme of Work: Appendix 1C

Appendix 1C

Submitted by the Government of Portugal*

In the annex of this document, submitted by Portugal (holding the Presidency of the Council of the European Union), contains amendment proposals to Annex IC in order to adapt the European Union specifications on the smart tachograph to the AETR legal framework.

*The modifications to this document are marked in strikethrough for deleted and in bold for new text.

Annex

INTRODUCTION	17
1. DEFINITIONS	19
2. GENERAL CHARACTERISTICS AND FUNCTIONS OF THE CONTROL DEVICE	26
2.1. <i>General characteristics</i>	26
2.2. <i>Functions</i>	26
2.3. <i>Modes of operation</i>	27
2.4. <i>Security</i>	29
3. CONSTRUCTION AND FUNCTIONAL REQUIREMENTS FOR CONTROL DEVICE	29
3.1. <i>Monitoring cards insertion and withdrawal</i>	29
3.2. <i>Speed, position and distance measurement</i>	30
3.3. <i>Time measurement</i>	31
3.4. <i>Monitoring driver activities</i>	32
3.5. <i>Monitoring driving status</i>	32
3.6. <i>Drivers entries</i>	32
3.7. <i>Company locks management</i>	36
3.8. <i>Monitoring control activities</i>	36
3.9. <i>Detection of events and/or faults</i>	36
3.10. <i>"Built-in and self-tests"</i>	39
3.11. <i>Reading from data memory</i>	40
3.12. <i>Recording and storing in the data memory</i>	40
3.14. <i>Recording and storing on tachograph cards</i>	52
3.15. <i>Displaying</i>	54
3.16. <i>Printing</i>	56
3.17. <i>Warnings</i>	57
3.18. <i>Data downloading to external media</i>	57
3.19. <i>Remote communication for targeted roadside checks</i>	58
3.20. <i>Output dData exchanges withto additional external devices</i>	58
3.21. <i>Calibration</i>	60
3.22. <i>Roadside calibration checking</i>	61
3.23. <i>Time adjustment</i>	61
3.24. <i>Performance characteristics</i>	62
3.25. <i>Materials</i>	63
3.26. <i>Markings</i>	63
3.27. <i>Monitoring border crossings</i>	63
3.28 Software update	64
4. CONSTRUCTION AND FUNCTIONAL REQUIREMENTS FOR TACHOGRAPH CARDS	64
4.1. <i>Visible data</i>	64

4.2.	<i>Security</i>	67
4.3.	<i>Standards</i>	68
4.4.	<i>Environmental and electrical specifications</i>	68
4.5.	<i>Data storage</i>	68
5.	INSTALLATION OF THE CONTROL DEVICE	89
5.1.	<i>Installation</i>	89
5.2.	<i>Installation plaque</i>	90
5.3.	<i>Sealing</i>	91
6.	CHECKS, INSPECTIONS AND REPAIRS	92
6.1.	<i>Approval of fitters, workshops and vehicle manufacturers</i>	92
6.2.	<i>Check of new or repaired instrumentscomponents</i>	92
6.3.	<i>Installation inspection</i>	93
6.4.	<i>Periodic inspections</i>	93
6.5.	<i>Measurement of errors</i>	94
6.6.	<i>Repairs</i>	94
7.	CARD ISSUING	94
8.	TYPE APPROVAL OF CONTROL DEVICES AND TACHOGRAPH CARDS	95
8.1.	<i>General points</i>	95
8.2.	<i>Security certificate</i>	96
8.3.	<i>Functional certificate</i>	96
8.4.	<i>Interoperability certificate</i>	97
8.5.	<i>Type approval certificate</i>	98
	APPENDIX: APPROVAL MARK AND CERTIFICATE	100
I.	APPROVAL MARK	100
II.	APPROVAL CERTIFICATE FOR ANALOGUE TACHOGRAPHS	102
III.	APPROVAL CERTIFICATE FOR DIGITAL TACHOGRAPHS	103
IV.	APPROVAL CERTIFICATE FOR SMART TACHOGRAPHS	104
	SUB-APPENDIX 1. DATA DICTIONARY	105
1.	INTRODUCTION	113
1.1	<i>Approach for definitions of data types</i>	113
1.2.	<i>References</i>	113
2.	DATA TYPE DEFINITIONS	114
2.1.	<i>ActivityChangeInfo</i>	114
2.2.	<i>Address</i>	116
2.3.	<i>AESKey</i>	116
2.4.	<i>AES128Key</i>	116
2.5.	<i>AES192Key</i>	116
2.6.	<i>AES256Key</i>	117
2.7.	<i>BCDString</i>	117

2.8.	<i>CalibrationPurpose</i>	117
2.9.	<i>CardActivityDailyRecord</i>	118
2.10.	<i>CardActivityLengthRange</i>	118
2.11.	<i>CardApprovalNumber</i>	119
2.11a.	<i>CardBorderCrossings</i>	119
2.11b.	<i>CardBorderCrossingRecordd</i>	119
2.12.	<i>CardCertificate</i>	120
2.13.	<i>CardChipIdentification</i>	120
2.14.	<i>CardConsecutiveIndex</i>	120
2.15.	<i>CardControlActivityDataRecord</i>	120
2.16.	<i>CardCurrentUse</i>	121
2.17.	<i>CardDriverActivity</i>	121
2.18.	<i>CardDrivingLicenceInformation</i>	122
2.19.	<i>CardEventData</i>	122
2.20.	<i>CardEventRecord</i>	123
2.21.	<i>CardFaultData</i>	123
2.22.	<i>CardFaultRecord</i>	124
2.23.	<i>CardIccIdentification</i>	124
2.24.	<i>CardIdentification</i>	125
2.24a.	<i>CardLoadTypeEntries</i>	125
2.25.	<i>CardMACertificate</i>	127
2.26.	<i>CardNumber</i>	127
2.26a.	<i>CardPlaceAuthDailyWorkPeriod</i>	127
2.27.	<i>CardPlaceDailyWorkPeriod</i>	128
2.28.	<i>CardPrivateKey</i>	128
2.29.	<i>CardPublicKey</i>	128
2.30.	<i>CardRenewalIndex</i>	129
2.31.	<i>CardReplacementIndex</i>	129
2.32.	<i>CardSignCertificate</i>	129
2.33.	<i>CardSlotNumber</i>	129
2.34.	<i>CardSlotsStatus</i>	129
2.35.	<i>CardSlotsStatusRecordArray</i>	130
2.36.	<i>CardStructureVersion</i>	130
2.37.	<i>CardVehicleRecord</i>	130
2.38.	<i>CardVehiclesUsed</i>	131
2.39.	<i>CardVehicleUnitRecord</i>	132
2.40.	<i>CardVehicleUnitsUsed</i>	132
2.41.	<i>Certificate</i>	133
2.42.	<i>CertificateContent</i>	133
2.43.	<i>CertificateHolderAuthorisation</i>	133
2.44.	<i>CertificateRequestID</i>	134

2.45.	<i>CertificationAuthorityKID</i>	135
2.46.	<i>CompanyActivityData</i>	135
2.47.	<i>CompanyActivityType</i>	136
2.48.	<i>CompanyCardApplicationIdentification</i>	136
2.48a.	<i>CompanyCardApplicationIdentificationV2</i>	136
2.49.	<i>CompanyCardHolderIdentification</i>	137
2.50.	<i>ControlCardApplicationIdentification</i>	137
2.50a.	<i>ControlCardApplicationIdentificationV2</i>	137
2.51.	<i>ControlCardControlActivityData</i>	138
2.52.	<i>ControlCardHolderIdentification</i>	139
2.53.	<i>ControlType</i>	139
2.54.	<i>CurrentDateTime</i>	140
2.55.	<i>CurrentDateTimeRecordArray</i>	140
2.56.	<i>DailyPresenceCounter</i>	141
2.57.	<i>Datef</i>	141
2.58.	<i>DateOfDayDownloaded</i>	141
2.59.	<i>DateOfDayDownloadedRecordArray</i>	141
2.60.	<i>Distance</i>	142
2.60a.	<i>DownloadInterfaceVersion</i>	142
2.61.	<i>DriverCardApplicationIdentification</i>	142
2.61a.	<i>DriverCardApplicationIdentificationV2</i>	143
2.62.	<i>DriverCardHolderIdentification</i>	144
2.63.	<i>DSRCSecurityData</i>	144
2.64.	<i>EGFCertificate</i>	145
2.65.	<i>EmbedderIcAssemblerId</i>	145
2.66.	<i>EntryTypeDailyWorkPeriod</i>	146
2.67.	<i>EquipmentType</i>	146
2.68.	<i>EuropeanPublicKey</i>	147
2.69.	<i>EventFaultRecordPurpose</i>	148
2.70.	<i>EventFaultType</i>	148
2.71.	<i>ExtendedSealIdentifier</i>	153
2.72.	<i>ExtendedSerialNumber</i>	154
2.73.	<i>FullCardNumber</i>	155
2.74.	<i>FullCardNumberAndGeneration</i>	155
2.75.	<i>Generation</i>	155
2.76.	<i>GeoCoordinates</i>	155
2.77.	<i>GNSSAccuracy</i>	156
2.78.	<i>GNSSAccumulatedDriving</i>	156
2.79.	<i>GNSSContinousDrivingRecordGNSSAccumulatedDrivingRecord</i>	156
2.79a.	<i>GNSSAuthAccumulatedDriving</i>	157
2.80.	<i>GNSSPlaceRecord</i>	158

2.81.	<i>HighResOdometer</i>	159
2.82.	<i>HighResTripDistance</i>	159
2.83.	<i>HolderName</i>	159
2.84.	<i>Reserved for future useInternalGNSSReceiver</i>	159
2.85.	<i>K-ConstantOfRecordingEquipment</i>	159
2.86.	<i>KeyIdentifier</i>	160
2.87.	<i>KMWCKey</i>	160
2.88.	<i>Language</i>	160
2.89.	<i>LastCardDownload</i>	160
2.89a.	<i>LengthOfFollowingData</i>	161
2.90.	<i>LinkCertificate</i>	161
2.90a.	<i>LoadType</i>	161
2.91.	<i>L-TyreCircumference</i>	161
2.92.	<i>MAC</i>	161
2.93.	<i>ManualInputFlag</i>	162
2.94.	<i>ManufacturerCode</i>	162
2.95.	<i>ManufacturerSpecificEventFaultData</i>	162
2.96.	<i>MemberStateCertificate</i>	162
2.97.	<i>MemberStateCertificateRecordArray</i>	162
2.98.	<i>MemberStatePublicKey</i>	163
2.99.	<i>Name</i>	163
2.100.	<i>NationAlpha</i>	163
2.101.	<i>NationNumeric</i>	164
2.101a.	<i>NoOfBorderCrossingRecords</i>	164
2.102.	<i>NoOfCalibrationRecords</i>	164
2.103.	<i>NoOfCalibrationsSinceDownload</i>	164
2.104.	<i>NoOfCardPlaceRecords</i>	164
2.106.	<i>NoOfCardVehicleUnitRecords</i>	165
2.107.	<i>NoOfCompanyActivityRecords</i>	165
2.108.	<i>NoOfControlActivityRecords</i>	165
2.109.	<i>NoOfEventsPerType</i>	165
2.110.	<i>NoOfFaultsPerType</i>	165
2.111.	<i>NoOfGNSSADRecords</i>	166
2.111a.	<i>NoOfLoadUnloadRecords</i>	166
2.112.	<i>NoOfSpecificConditionRecords</i>	166
2.112a.	<i>NoOfLoadTypeEntryRecords</i>	166
2.113.	<i>OdometerShort</i>	166
2.114.	<i>OdometerValueMidnight</i>	166
2.114a.	<i>OperationType</i>	167
2.115.	<i>OdometerValueMidnightRecordArray</i>	167
2.116.	<i>OverspeedNumber</i>	167

2.116a. <i>PlaceAuthRecord</i>	168
2.117. <i>PlaceRecord</i>	168
2.117a <i>PositionAuthenticationStatus</i>	169
2.118. <i>PreviousVehicleInfo</i>	169
2.119. <i>PublicKey</i>	170
2.120. <i>RecordType</i>	170
2.121. <i>RegionAlpha</i>	172
2.122. <i>RegionNumeric</i>	172
2.123. <i>RemoteCommunicationModuleSerialNumber</i>	173
2.124. <i>RSAPublicModulus</i>	173
2.125. <i>RSAPublicExponent</i>	173
2.126. <i>RSAPrivateExponent</i>	174
2.127. <i>RtmData</i>	174
2.128. <i>SealDataCard</i>	174
2.129. <i>SealDataVu</i>	174
2.130. <i>SealRecord</i>	174
2.131. <i>SensorApprovalNumber</i>	175
2.132. <i>SensorExternalGNSSApprovalNumber</i>	175
2.133. <i>SensorExternalGNSSCoupledRecord</i>	175
2.134. <i>SensorExternalGNSSIdentification</i>	176
2.135. <i>SensorExternalGNSSInstallation</i>	176
2.136. <i>SensorExternalGNSSOSIdentifier</i>	177
2.137. <i>SensorExternalGNSSSCIdentifier</i>	177
2.138. <i>SensorGNSSCouplingDate</i>	177
2.139. <i>SensorGNSSSerialNumber</i>	177
2.140. <i>SensorIdentification</i>	177
2.141. <i>SensorInstallation</i>	178
2.142. <i>SensorInstallationSecData</i>	178
2.143. <i>SensorOSIdentifier</i>	179
2.144. <i>SensorPaired</i>	179
2.145. <i>SensorPairedRecord</i>	179
2.146. <i>SensorPairingDate</i>	180
2.147. <i>SensorSCIdentifier</i>	180
2.148. <i>SensorSerialNumber</i>	180
2.149. <i>Signature</i>	180
2.150. <i>SignatureRecordArray</i>	180
2.151. <i>SimilarEventsNumber</i>	181
2.152. <i>SpecificConditionRecord</i>	181
2.153. <i>SpecificConditions</i>	181
2.154. <i>SpecificConditionType</i>	182
2.155. <i>Speed</i>	182

2.156. <i>SpeedAuthorised</i>	182
2.157. <i>SpeedAverage</i>	183
2.158. <i>SpeedMax</i>	183
2.158a. <i>TachographCardsGen1Suppression</i>	183
2.159. <i>TachographPayload</i>	183
2.160. <i>Reserved for future use</i>	183
2.161. <i>TDesSessionKey</i>	184
2.162. <i>TimeReal</i>	184
2.163. <i>TyreSize</i>	184
2.164. <i>VehicleIdentificationNumber</i>	184
2.165. <i>VehicleIdentificationNumberRecordArray</i>	184
2.166. <i>VehicleRegistrationIdentification</i>	185
2.166a. <i>VehicleRegistrationIdentificationRecordArray</i>	185
2.167. <i>VehicleRegistrationNumber</i>	186
2.168. <i>VehicleRegistrationNumberRecordArray</i>	186
2.169. <i>VuAbility</i>	186
2.170. <i>VuActivityDailyData</i>	187
2.171. <i>VuActivityDailyRecordArray</i>	187
2.172. <i>VuApprovalNumber</i>	187
2.173. <i>VuCalibrationData</i>	188
2.174. <i>VuCalibrationRecord</i>	188
2.175. <i>VuCalibrationRecordArray</i>	191
2.176. <i>VuCardIWData</i>	191
2.177. <i>VuCardIWRecord</i>	192
2.178. <i>VuCardIWRecordArray</i>	193
2.179. <i>VuCardRecord</i>	194
2.180. <i>VuCardRecordArray</i>	194
2.181. <i>VuCertificate</i>	195
2.182. <i>VuCertificateRecordArray</i>	195
2.183. <i>VuCompanyLocksData</i>	195
2.184. <i>VuCompanyLocksRecord</i>	195
2.185. <i>VuCompanyLocksRecordArray</i>	196
2.185a. <i>VuConfigurationLengthRange</i>	197
2.186. <i>VuControlActivityData</i>	197
2.187. <i>VuControlActivityRecord</i>	197
2.188. <i>VuControlActivityRecordArray</i>	198
2.189. <i>VuDataBlockCounter</i>	198
2.190. <i>VuDetailedSpeedBlock</i>	199
2.191. <i>VuDetailedSpeedBlockRecordArray</i>	199
2.192. <i>VuDetailedSpeedData</i>	199
2.192a. <i>VuDigitalMapVersion</i>	200

2.193.	<i>VuDownloadablePeriod</i>	200
2.194.	<i>VuDownloadablePeriodRecordArray</i>	200
2.195.	<i>VuDownloadActivityData</i>	200
2.196.	<i>VuDownloadActivityDataRecordArray</i>	201
2.197.	<i>VuEventData</i>	201
2.198.	<i>VuEventRecord</i>	202
2.199.	<i>VuEventRecordArray</i>	203
2.200.	<i>VuFaultData</i>	204
2.201.	<i>VuFaultRecord</i>	204
2.202.	<i>VuFaultRecordArray</i>	205
2.203.	<i>VuGNSSADRecord</i>	206
2.203a.	<i>VuBorderCrossingRecord</i>	207
2.203b.	<i>VuBorderCrossingRecordArray</i>	207
2.204.	<i>VuGNSSADRecordArray</i>	208
2.204a.	<i>VuGnssMaximalTimeDifference</i>	208
2.205.	<i>VuIdentification</i>	209
2.206.	<i>VuIdentificationRecordArray</i>	210
2.207.	<i>VuITSConsentRecord</i>	210
2.208.	<i>VuITSConsentRecordArray</i>	211
2.208a.	<i>VuLoadUnloadRecord</i>	211
2.209.	<i>VuManufacturerAddress</i>	212
2.210.	<i>VuManufacturerName</i>	212
2.211.	<i>VuManufacturingDate</i>	212
2.212.	<i>VuOverSpeedingControlData</i>	212
2.213.	<i>VuOverSpeedingControlDataRecordArray</i>	213
2.214.	<i>VuOverSpeedingEventData</i>	213
2.215.	<i>VuOverSpeedingEventRecord</i>	214
2.216.	<i>VuOverSpeedingEventRecordArray</i>	215
2.217.	<i>VuPartNumber</i>	215
2.218.	<i>VuPlaceDailyWorkPeriodData</i>	215
2.219.	<i>VuPlaceDailyWorkPeriodRecord</i>	216
2.220.	<i>VuPlaceDailyWorkPeriodRecordArray</i>	217
2.221.	<i>VuPrivateKey</i>	217
2.222.	<i>VuPublicKey</i>	217
2.222a.	<i>VuRtcTime</i>	217
2.223.	<i>VuSerialNumber</i>	217
2.224.	<i>VuSoftInstallationDate</i>	218
2.225.	<i>VuSoftwareIdentification</i>	218
2.226.	<i>VuSoftwareVersion</i>	218
2.227.	<i>VuSpecificConditionData</i>	218
2.228.	<i>VuSpecificConditionRecordArray</i>	218

2.229.	<i>VuTimeAdjustmentData</i>	219
2.230.	<i>Reserved for future use</i>	219
2.230.	<i>VuTimeAdjustmentGNSSRecord</i>	219
2.231.	<i>Reserved for future use</i>	219
2.232.	<i>VuTimeAdjustmentRecord</i>	219
2.233.	<i>VuTimeAdjustmentRecordArray</i>	221
2.234.	<i>WorkshopCardApplicationIdentification</i>	221
2.234a.	<i>WorkshopCardApplicationIdentificationV2</i>	222
2.234c.	<i>WorkshopCardCalibrationAddDataRecord</i>	223
2.235.	<i>WorkshopCardCalibrationData</i>	224
2.236.	<i>WorkshopCardCalibrationRecord</i>	224
2.237.	<i>WorkshopCardHolderIdentification</i>	226
2.238.	<i>WorkshopCardPIN</i>	227
2.239.	<i>W-VehicleCharacteristicConstant</i>	227
2.240.	<i>VuPowerSupplyInterruptionRecord</i>	227
2.241.	<i>VuPowerSupplyInterruptionRecordArray</i>	228
2.242.	<i>VuSensorExternalGNSSCoupledRecordArray</i>	228
2.243.	<i>VuSensorPairedRecordArray</i>	229
3.	VALUE AND SIZE RANGE DEFINITIONS	229
4.	CHARACTER SETS	229
5.	ENCODING	230
6.	OBJECT IDENTIFIERS AND APPLICATION IDENTIFIERS	230
6.1.	<i>Object Identifiers</i>	230
6.2.	<i>Application identifiers</i>	231
	SUB-APPENDIX 2. TACHOGRAPH CARDS SPECIFICATION	232
1.	INTRODUCTION	234
1.1.	<i>Abbreviations</i>	234
1.2.	<i>References</i>	235
2.	ELECTRICAL AND PHYSICAL CHARACTERISTICS	235
2.1.	<i>Supply Voltage and Current Consumption</i>	236
2.2.	<i>Programming Voltage Vpp</i>	236
2.3.	<i>Clock generation and Frequency</i>	236
2.4.	<i>I/O Contact</i>	236
2.5.	<i>States of the Card</i>	236
3.	HARDWARE AND COMMUNICATION	237
3.1.	<i>Introduction</i>	237
3.2.	<i>Transmission Protocol</i>	237
3.3.	<i>Access rules</i>	239
3.4.	<i>Commands and error codes overview</i>	241
3.5.	<i>Command descriptions</i>	244

4.	TACHOGRAPH CARDS STRUCTURE	275
4.1.	<i>Master File MF</i>	275
4.2.	<i>Driver card applications</i>	277
4.3.	<i>Workshop card applications</i>	286
4.4.	<i>Control card applications</i>	299
4.5.	<i>Company card applications</i>	304
	SUB-APPENDIX 3. PICTOGRAMS	309
	SUB-APPENDIX 4. PRINTOUTS	312
1.	GENERALITIES	313
2.	DATA BLOCKS SPECIFICATION	313
	SUB-APPENDIX 5. DISPLAY	326
	SUB-APPENDIX 6. FRONT CONNECTOR FOR CALIBRATION AND DOWNLOAD	328
1.	HARDWARE	329
1.1.	<i>Connector</i>	329
1.2.	<i>Contact allocation</i>	331
1.3.	<i>Block diagram</i>	331
2.	DOWNLOADING INTERFACE	331
3.	CALIBRATION INTERFACE	332
	SUB-APPENDIX 7. DATA DOWNLOADING PROTOCOLS	334
1.	INTRODUCTION	335
1.1.	<i>Scope</i>	335
1.2.	<i>Acronyms and notations</i>	335
2.	V.U. DATA DOWNLOADING	336
2.1.	<i>Download procedure</i>	336
2.2.	<i>Data download protocol</i>	336
2.3.	<i>ESM File storage</i>	356
3.	TACHOGRAPH CARDS DOWNLOADING PROTOCOL	357
3.1.	<i>Scope</i>	357
3.2.	<i>Definitions</i>	357
3.3.	<i>Card Downloading</i>	357
3.4.	<i>Data storage format</i>	360
4.	DOWNLOADING A TACHOGRAPH CARD VIA A VEHICLE UNIT	361
	SUB-APPENDIX 8. CALIBRATION PROTOCOL	363
1.	INTRODUCTION	364
2.	TERMS, DEFINITIONS AND REFERENCES	366
3.	OVERVIEW OF SERVICES	367
3.1.	<i>Services available</i>	367
3.2.	<i>Response codes</i>	368

4.	COMMUNICATION SERVICES	368
4.1.	<i>StartCommunication Service</i>	368
4.2.	<i>StopCommunication Service</i>	370
4.3.	<i>TesterPresent Service</i>	371
5.	MANAGEMENT SERVICES	373
5.1.	<i>StartDiagnosticSession service</i>	373
5.2.	<i>SecurityAccess service</i>	375
6.	DATA TRANSMISSION SERVICES	378
6.1.	<i>ReadDataByIdentifier service</i>	378
6.2.	<i>WriteDataByIdentifier service</i>	381
7.	CONTROL OF TEST PULSES – INPUT/OUTPUT CONTROL FUNCTIONAL UNIT	382
7.1.	<i>InputOutputControlByIdentifier service</i>	382
8.	ROUTINE CONTROL SERVICE (TIME ADJUSTMENT)	385
8.1	<i>Message description</i>	384
8.2	<i>Message format</i>	384
9.	DATARECORDS FORMATS	387
9.1.	<i>Transmitted parameter ranges</i>	387
9.2.	<i>dataRecords formats</i>	388
	SUB-APPENDIX 9. TYPE APPROVAL	392
1.	INTRODUCTION	393
1.1.	<i>Type approval</i>	393
1.2.	<i>References</i>	393
2.	VEHICLE UNIT FUNCTIONAL TESTS	395
3.	MOTION SENSOR FUNCTIONAL TESTS	399
4.	TACHOGRAPH CARDS FUNCTIONAL TESTS	402
5.	EXTERNAL GNSS FACILITY TESTS	409
6.	EXTERNAL REMOTE COMMUNICATION FACILITY TESTS	411
7.	PAPER FUNCTIONAL TESTS	413
8.	INTEROPERABILITY TESTS	415
9.	OSNMA TESTS	414
	SUB-APPENDIX 10. SECURITY REQUIREMENTS	420
	SUB-APPENDIX 11. COMMON SECURITY MECHANISMS	421
	PREAMBLE	423
	PART A FIRST-GENERATION TACHOGRAPH SYSTEM	423
1.	INTRODUCTION	423
1.1.	<i>References</i>	423
1.2.	<i>Notations and abbreviated terms</i>	424

2.	CRYPTOGRAPHIC SYSTEMS AND ALGORITHMS.....	425
2.1.	<i>Cryptographic systems.....</i>	425
2.2.	<i>Cryptographic algorithms.....</i>	425
3.	KEYS AND CERTIFICATES.....	426
3.1.	<i>Keys generation and distribution.....</i>	426
3.2.	<i>Keys.....</i>	428
3.3.	<i>Certificates.....</i>	428
4.	MUTUAL AUTHENTICATION MECHANISM.....	431
5.	VU-CARDS DATA TRANSFER CONFIDENTIALITY, INTEGRITY AND AUTHENTICATION MECHANISMS.....	435
5.1.	<i>Secure Messaging.....</i>	435
5.2.	<i>Treatment of Secure Messaging errors.....</i>	436
5.3.	<i>Algorithm to compute Cryptographic Checksums.....</i>	437
5.4.	<i>Algorithm to compute cryptograms for confidentiality Dos.....</i>	437
6.	DATA DOWNLOAD DIGITAL SIGNATURE MECHANISMS.....	438
6.1.	<i>Signature generation.....</i>	438
6.2.	<i>Signature verification.....</i>	438
PART B SECOND-GENERATION TACHOGRAPH SYSTEM.....		440
7.	INTRODUCTION.....	440
7.1.	<i>References.....</i>	440
7.2.	<i>Notations and Abbreviations.....</i>	440
7.3.	<i>Definitions.....</i>	442
8.	CRYPTOGRAPHIC SYSTEMS AND ALGORITHMS.....	442
8.1.	<i>Cryptographic Systems.....</i>	442
8.2.	<i>Cryptographic Algorithms.....</i>	443
9.	KEYS AND CERTIFICATES.....	444
9.1.	<i>Asymmetric Key Pairs and Public Key Certificates.....</i>	444
9.2.	<i>Symmetric Keys.....</i>	452
9.3.	<i>Certificates.....</i>	459
10.	VU- CARD MUTUAL AUTHENTICATION AND SECURE MESSAGING.....	462
10.1.	<i>General.....</i>	462
10.2.	<i>Mutual Certificate Chain Verification.....</i>	463
10.3.	<i>VU Authentication.....</i>	469
10.4.	<i>Chip Authentication and Session Key Agreement.....</i>	470
10.5.	<i>Secure Messaging.....</i>	472
11.	VU – EXTERNAL GNSS FACILITY COUPLING, MUTUAL AUTHENTICATION AND SECURE MESSAGING.....	477
11.1.	<i>General.....</i>	477
11.2.	<i>VU and External GNSS Facility Coupling.....</i>	478
11.3.	<i>Mutual Certificate Chain Verification.....</i>	478
11.4.	<i>VU Authentication, Chip Authentication and Session Key Agreement.....</i>	480

11.5.	<i>Secure messaging</i>	480
12.	VU – MOTION SENSOR PAIRING AND COMMUNICATION	481
12.1.	<i>General</i>	481
12.2.	<i>VU – Motion Sensor Pairing Using Different Key Generations</i>	481
12.3.	<i>VU – Motion Sensor Pairing and Communication using AES</i>	482
12.4.	<i>VU – Motion Sensor Pairing For Different Equipment Generations</i>	484
13.	SECURITY FOR REMOTE COMMUNICATION OVER DSRC	484
13.1.	<i>General</i>	484
13.2.	<i>Tachograph Payload Encryption and MAC Generation</i>	485
13.3.	<i>Verification and Decryption of Tachograph Payload</i>	486
14.	SIGNING DATA DOWNLOADS AND VERIFYING SIGNATURES	487
14.1.	<i>General</i>	487
14.2.	<i>Signature generation</i>	487
14.3.	<i>Signature verification</i>	488
SUB-APPENDIX 12. POSITIONING BASED ON GLOBAL NAVIGATION SATELLITE SYSTEM (GNSS)		490
1.	INTRODUCTION	491
1.1.	<i>Scope</i>	491
1.2.	<i>Acronyms and notations</i>	492
2.	BASIC CHARACTERISTICS OF THE GNSS RECEIVER	492
3.	SENTENCES PROVIDED BY THE GNSS RECEIVER	494
4.	VEHICLE UNIT WITH AN EXTERNAL GNSS FACILITY	497
4.1.	<i>Configuration</i>	497
4.2.	<i>Communication between the external GNSS facility and the vehicle unit</i>	498
4.3.	<i>Coupling, mutual authentication and session key agreement of the external GNSS facility with vehicle unit</i>	503
4.4.	<i>Error Handling</i>	503
5.	VEHICLE UNIT WITHOUT AN EXTERNAL GNSS FACILITY	505
5.1.	<i>Configuration</i>	505
5.2.	<i>Transfer of information from the GNSS receiver to the VU</i>	505
5.3.	<i>Transfer of information from the VU to the GNSS receiver</i>	505
5.4.	<i>Error Handling</i>	505
6.	POSITION DATA PROCESSING AND RECORDING BY THE VU	506
7.	GNSS TIME CONFLICT	507
8.	VEHICLE MOTION CONFLICT	507
SUB-APPENDIX 13. ITS INTERFACE		510
1.	INTRODUCTION	511
1.1	<i>Scope</i>	510
1.2	<i>Acronyms and definitions</i>	510
2.	REFERENCED STANDARDS	510

3.	ITS INTERFACE WORKING PRINCIPLES	510
3.1	<i>Communication technology</i>	<i>511</i>
3.2	<i>Available services.....</i>	<i>511</i>
3.3	<i>Access through the ITS interface.....</i>	<i>512</i>
3.4	<i>Data available and need of driver consent</i>	<i>513</i>
4.	LIST OF DATA AVAILABLE THROUGH THE ITS INTERFACE AND PERSONAL/ NOT PERSONAL CLASSIFICATION	514
SUB-APPENDIX 14. REMOTE COMMUNICATION FUNCTION		552
1.	INTRODUCTION	554
2.	SCOPE	555
3.	ACRONYMS, DEFINITIONS AND NOTATIONS	556
4.	OPERATIONAL SCENARIOS	558
4.1.	<i>Overview</i>	<i>558</i>
4.2.	<i>Security/Integrity</i>	<i>560</i>
5.	REMOTE COMMUNICATION DESIGN AND PROTOCOLS.....	560
5.1.	<i>Design</i>	<i>560</i>
5.2.	<i>Workflow</i>	<i>564</i>
5.3.	<i>DSRC Physical interface parameters for remote communication</i>	<i>565</i>
5.4.	<i>DSRC Protocol requirements for RTM</i>	<i>570</i>
5.5.	<i>Support for Directive 2015/71/EC (Text deleted) Reserved for future use.....</i>	<i>595</i>
5.6.	<i>Data transfer between the DSRC_VU and VU.....</i>	<i>595</i>
5.7.	<i>Error handling</i>	<i>597</i>
6.	COMMISSIONING AND PERIODIC INSPECTION TESTS FOR THE REMOTE COMMUNICATION FUNCTION	599
6.1.	<i>General</i>	<i>599</i>
6.2.	<i>ECHO</i>	<i>599</i>
6.3.	<i>Tests to validate the secure data content</i>	<i>600</i>
SUB-APPENDIX 15. MIGRATION: MANAGING THE CO-EXISTENCE OF EQUIPMENT GENERATIONS		607
1.	DEFINITIONS.....	608
2.	GENERAL PROVISIONS.....	608
2.1.	<i>Overview of the transition</i>	<i>608</i>
2.2.	<i>Interoperability between VU and cards</i>	<i>608</i>
2.3.	<i>Interoperability between VU and MS.....</i>	<i>609</i>
2.4.	<i>Interoperability between vehicle units, tachograph cards and equipment for data download.....</i>	<i>609</i>
2.5.	<i>Interoperability between VU and calibration equipment</i>	<i>610</i>
3.	MAIN STEPS DURING THE PERIOD BEFORE THE INTRODUCTION DATE	610
4.	PROVISIONS FOR THE PERIOD AFTER THE INTRODUCTION DATE.....	611
SUB-APPENDIX 16. ADAPTOR FOR M1 AND N1 CATEGORY VEHICLES		612
1.	ABBREVIATIONS AND REFERENCE DOCUMENTS.....	613

1.1.	<i>Abbreviations</i>	613
1.2.	<i>Reference standards</i>	613
2.	GENERAL CHARACTERISTICS AND FUNCTIONS OF THE ADAPTOR	613
2.1.	<i>Adaptor general description</i>	613
2.2.	<i>Functions</i>	613
2.3.	<i>Security</i>	613
3.	REQUIREMENTS FOR THE RECORDING EQUIPMENT CONTROL DEVICE WHEN AN ADAPTOR IS INSTALLED	614
4.	CONSTRUCTION AND FUNCTIONAL REQUIREMENTS FOR THE ADAPTOR	614
4.1.	<i>Interfacing and adapting incoming speed pulses</i>	614
4.2.	<i>Inducing the incoming pulses to the embedded motion sensor</i>	615
4.3.	<i>Embedded motion sensor</i>	615
4.4.	<i>Security requirements</i>	615
4.5.	<i>Performance characteristics</i>	615
4.6.	<i>Materials</i>	616
5.	INSTALLATION OF THE CONTROL DEVICE WHEN AN ADAPTOR IS USED	616
5.1.	<i>Installation</i>	616
5.2.	<i>Sealing</i>	616
6.	CHECKS, INSPECTIONS AND REPAIRS	617
6.1.	<i>Periodic inspections</i>	617
7.	TYPE APPROVAL OF CONTROL DEVICE WHEN AN ADAPTOR IS USED	617
7.1.	<i>General points</i>	617
7.2.	<i>Functional certificate</i>	617

Introduction

First generation digital tachograph system is deployed ~~since 1 May 2006~~ **on the territory of the Contracting Parties**. It may be used until its end of life for domestic transportation. ~~For international transportation instead, 15 years after the entry into force of this Commission Regulation, all vehicles shall be equipped with a compliant second generation smart tachograph, introduced by this Regulation.~~

First generation tachograph system complies with Appendix 1B to this Agreement, while second generation tachograph system, also called smart tachograph system, complies with this sub-Appendix. This ~~Annex~~ **Appendix** contains second generation recording equipment **control device** and tachograph cards requirements.

Starting from its introduction date, second generation ~~version 2 recording equipment control devices~~ **version 2 recording equipment control devices** shall be installed in vehicles **operating in the scope of this Agreement** ~~registered for the first time~~, and second generation **version 2** tachograph cards shall be issued. In order to foster a smooth introduction of the second generation tachograph system,

- second generation **version 2** tachograph cards shall be designed to be also used in first generation **and second generation version 1** vehicle units,
- replacement of valid first generation **and second generation version 1** tachograph cards at the introduction date shall not be requested.

This will allow drivers to keep their unique driver card and use both systems with it.

Second generation ~~recording equipment control device~~ **control device** shall however only be calibrated using second generation workshop cards.

This ~~Annex~~ **Appendix** contains all requirements related to the interoperability between the first and the second generation tachograph systems.

~~Appendix~~ **Sub-appendix** 15 contains additional details about how the co-existence of the two systems ~~generations~~ shall be managed, **including the different versions of the second generation.**

List of ~~Appendixes~~ **Sub-appendixes**

- App 1: DATA DICTIONARY
- App 2: TACHOGRAPH CARDS SPECIFICATION
- App 3: PICTOGRAMS
- App 4: PRINTOUTS
- App 5: DISPLAY
- App 6: FRONT CONNECTOR FOR CALIBRATION AND DOWNLOAD
- App 7: DATA DOWNLOADING PROTOCOLS
- App 8: CALIBRATION PROTOCOL
- App 9: TYPE APPROVAL AND LIST OF MINIMUM REQUIRED TESTS
- App 10: SECURITY REQUIREMENTS
- App 11: COMMON SECURITY MECHANISMS
- App 12: POSITIONING BASED ON GLOBAL NAVIGATION SATELLITE SYSTEM (GNSS)
- App 13: ITS INTERFACE

- App 14: REMOTE COMMUNICATION FUNCTION
- App 15: MIGRATION: MANAGING THE CO-EXISTENCE OF EQUIPMENT GENERATIONS
- App 16: ADAPTOR FOR M 1 AND N1 CATEGORY VEHICLES

1. Definitions

In this ~~Annex~~ **Appendix**:

- (a) “activation” means:
the phase in which the tachograph becomes fully operational and implements all functions, including security functions, through the use of a workshop card;
- (b) “authentication” means:
a function intended to establish and verify a claimed identity;
- (c) “authenticity” means:
the property that information is coming from a party whose identity can be verified;
- (d) “built-in-test (BIT)” means:
tests run at request, triggered by the operator or by external equipment;
- (e) “calendar day” means:
a day ranging from 00.00 hours to 24.00 hours. All calendar days relate to UTC time (Universal Time Co-ordinated);
- (f) “calibration” of a smart tachograph means:
updating or confirming vehicle parameters to be held in the data memory. Vehicle parameters include vehicle identification (VIN, VRN and registering ~~Member States~~ **Contracting Party**) and vehicle characteristics (w, k, l, tyre size, speed limiting device setting (if applicable), current UTC time, current odometer value, **by-default load type**); during the calibration of a ~~recording equipment~~ **control device**, the types and identifiers of all type approval relevant seals in place shall also be stored in the data memory;
any update or confirmation of UTC time only, shall be considered as a time adjustment and not as a calibration, provided it does not contradict ~~Requirement~~ **requirement 409**;
calibrating a ~~recording equipment~~ **control device** requires the use of a workshop card;
- (g) “card number” means:
a 16 alpha-numerical characters number that uniquely identifies a tachograph card within a ~~Member States~~ **Contracting Party**. The card number includes **an identification, which consists in a driver identification, or in a card owner identification together with a card consecutive index** ~~(if applicable)~~, a card replacement index and a card renewal index;
a card is therefore uniquely identified by the code of the issuing ~~Member States~~ **Contracting Party** and the card number;
- (h) “card consecutive index” means:
the 14th alpha-numerical character of a card number that is used to differentiate the different cards issued to a company, a workshop or a control authority entitled to be issued several tachograph cards. The company, the workshop or the control authority is uniquely identified by the 13 first characters of the card number;
- (i) “card renewal index” means:
the 16th alpha-numerical character of a card number which is incremented each time a tachograph card **corresponding to a given identification, i.e. driver identification or owner identification together with consecutive index**, is renewed;
- (j) “card replacement index” means:
the 15th alpha-numerical character of a card number which is incremented each time a tachograph card **corresponding to a given identification, i.e. driver identification or owner identification together with consecutive index**, is replaced;
- (k) “characteristic coefficient of the vehicle” means:

the numerical characteristic giving the value of the output signal emitted by the part of the vehicle linking it with the ~~recording equipment~~ **control device** (gearbox output shaft or axle) while the vehicle travels a distance of one kilometre under standard test conditions as defined under requirement 414. The characteristic coefficient is expressed in impulses per kilometre ($w = \dots \text{ imp/km}$);

(l) “company card” means:

a tachograph card issued by the authorities of a ~~Member States~~ **Contracting Party** to a transport undertaking needing to operate vehicles fitted with a tachograph, which identifies the transport undertaking and allows for the displaying, downloading and printing of the data, stored in the tachograph, which have been locked by that transport undertaking;

(m) “constant of the ~~recording equipment~~ **control device**” means:

the numerical characteristic giving the value of the input signal required to show and record a distance travelled of one kilometre; this constant shall be expressed in impulses per kilometre ($k = \dots \text{ imp/km}$);

(n) “continuous driving time” is computed within the ~~recording equipment~~ **control device** as¹:

the continuous driving time is computed as the current accumulated driving times of a particular driver, since the end of his last AVAILABILITY or BREAK/REST or UNKNOWN² period of 45 minutes or more (this period may have been split according to ~~Regulation (EC) N°. 561/2006~~. **this Agreement**). The computations involved take into account, as needed, past activities stored on the driver card. When the driver has not inserted his card, the computations involved are based on the data memory recordings related to the current period where no card was inserted and related to the relevant slot;

(o) “control card” means:

a tachograph card issued by the authorities of a ~~Member States~~ **Contracting Party** to a national competent control authority which identifies the control body and, optionally, the control officer, and which allows access to the data stored in the data memory or in the driver cards and, optionally, in the workshop cards for reading, printing and/or downloading;

It shall also give access to the roadside calibration checking function and to data on the remote early detection communication reader.

(p) “cumulative break time” is computed within the ~~recording equipment~~ **control device** as¹:

the cumulative break from driving time is computed as the current accumulated AVAILABILITY or BREAK/REST or UNKNOWN² times of 15 minutes or more of a particular driver, since the end of his last AVAILABILITY or BREAK/REST or UNKNOWN² period of 45 minutes or more (this period may have been split according to ~~Regulation (EC) N°. 561/2006~~. **this Agreement**).

The computations involved take into account, as needed, past activities stored on the driver card. Unknown periods of negative duration (start of unknown period > end of unknown period) due to time overlaps between two different ~~recording equipment~~ **control devices**, are not taken into account for the computation.

¹ This way of computing the continuous driving time and the cumulative break time serves into the ~~Recording Equipment Control device~~ for computing the continuous driving time warning. It does not prejudice the legal interpretation to be made of these times. Alternative ways of computing the continuous driving time and the cumulative break time may be used to replace these definitions if they have been made obsolete by updates in other relevant legislation.

² UNKNOWN periods correspond to periods where the driver’s card was not inserted in a ~~recording equipment~~ **control device** and for which no manual entry of driver activities was made.

When the driver has not inserted his card, the computations involved are based on the data memory recordings related to the current period where no card was inserted and related to the relevant slot

(q) “data memory” means:

an electronic data storage device built into the ~~recording equipment~~ **control device**;

(r) “digital signature” means:

data appended to, or a cryptographic transformation of, a block of data that allows the recipient of the block of data to prove the authenticity and integrity of the block of data;

(s) “downloading” means:

the copying, together with the digital signature, of a part, or of a complete set, of data files recorded in the data memory of the vehicle unit or in the memory of a tachograph card, provided that this process does not alter or delete any stored data;

Manufacturers of smart tachograph vehicle units and manufacturers of equipment designed and intended to download data files shall take all reasonable steps to ensure that the downloading of such data can be performed with the minimum delay by transport undertakings or drivers.

The downloading of the detailed speed file may not be necessary to establish compliance ~~Regulation (EC) No. 561/2006~~ with **this Agreement**, but may be used for other purposes such as accident investigation.

(t) “driver card” means:

a tachograph card, issued by the authorities of a ~~Member State~~ **Contracting Party** to a particular driver, which identifies the driver and allows for the storage of driver activity data;

(u) “effective circumference of the wheels” means:

the average of the distances travelled by each of the wheels moving the vehicle (driving wheels) in the course of one complete rotation. The measurement of these distances shall be made under standard test conditions as defined under requirement 414 and is expressed in the form “l = ... mm”. Vehicle manufacturers may replace the measurement of these distances by a theoretical calculation which takes into account the distribution of the weight on the axles, vehicle unladen in normal running order³, **namely with a coolant fluid, lubricants, fuel, tools, spare-wheel and driver**. The methods for such theoretical calculation are subject to approval by a competent ~~Member State~~ **Contracting Party** authority and can take place only before tachograph activation;

(v) “event” means:

an abnormal operation detected by the smart tachograph which may result from a fraud attempt;

(w) “external GNSS facility” means

a facility which contains the GNSS receiver when the vehicle unit is not a single unit as well as other components needed to protect the communication of position data to the rest of the vehicle unit;

(x) “fault” means:

abnormal operation detected by the smart tachograph which may come from an equipment malfunction or failure;

(y) “GNSS receiver” means:

an electronic device that receives and digitally processes the signals from one or more Global Navigation Satellite System(s) (GNSS in English) in order to provide position, speed and time information.

³ Regulation (EU) No. 1230/2012 relating to the masses and dimensions of certain categories of motor vehicles and the trailers and amending Directive 2007/46/EC as last amended.

(z) “installation” means:
the mounting of a tachograph in a vehicle;

(aa) ‘interoperability’ means:
the capacity of systems and the underlying business processes to exchange data and to share information;

(bb) ‘interface’ means:
a facility between systems which provides the media through which they can connect and interact;

(cc) “position” means:
geographical coordinates of the vehicle at a given time;

(dd) “motion sensor” means:
a part of the tachograph, providing a signal representative of vehicle speed and/or distance travelled;

(ee) “non valid card” means:
a card detected as faulty, or which ~~initial~~ authentication failed, or whose start of validity date is not yet reached, or which expiry date has passed;

a card is also considered as non-valid by the vehicle unit:

- **if a card with the same card issuing Contracting Party, the same identification, i.e. driver identification or owner identification together with consecutive index, and a higher renewal index has already been inserted in the vehicle unit, or**

- **if a card with the same card issuing Contracting Party, the same identification, i.e. driver identification or owner identification together with consecutive index and renewal index but with a higher replacement index has already been inserted in the vehicle unit;**

(ff) ‘open standard’ means:
a standard set out in a standard specification document available freely or at a nominal charge which it is permissible to copy, distribute or use for no fee or for a nominal fee.

(gg) “out of scope” means:
when the use of the ~~recording equipment control device~~ is not required, according to the provisions ~~Regulation (EC) N°. 561/2006~~ of **this Agreement**.

(hh) “over speeding” means:
exceeding the authorised speed of the vehicle, defined as any period of more than 60 seconds during which the vehicle’s measured speed exceeds the limit for setting the speed limitation device laid down in ~~Council Directive 92/6/EEC of 10 February 1992 on the installation and use of speed limitation devices for certain categories of motor vehicles in the Community⁴, as last amended-UNECE Regulation 89;~~

(ii) “periodic inspection” means:
a set of operations performed to check that the tachograph works properly, that its settings correspond to the vehicle parameters, and that no manipulation devices are attached to the tachograph;

(jj) “printer” means:
component of the ~~recording equipment control device~~ which provides printouts of stored data;

(kk) “remote early detection communication” means:

⁴ ~~OJ No. L057, 02/03/1992, p.0027-0028.~~

communication between the remote early detection communication facility and the remote early detection communication reader during targeted roadside checks with the aim of remotely detecting possible manipulation or misuse of ~~recording device~~ **control device**;

(ll) “remote communication **facility**”, “remote communication module” or “remote early detection facility” means:

the equipment of the vehicle unit which is used to perform targeted roadside checks;

(mm) “remote early detection communication reader” means:

the system used by control officers for targeted roadside checks.

(nn) “card renewal” means:

issue of a new tachograph card when an existing card reaches its expiry date, or is malfunctioning and has been returned to the issuing authority. ~~Renewal always implies the certainty that two valid cards do not co-exist;~~

(oo) “repair” means:

any repair of a motion sensor or of a vehicle unit or of a cable that requires the disconnection of its power supply, or its disconnection from other tachograph components, or the opening of the motion sensor or vehicle unit;

(pp) “card replacement” means:

issue of a **new** tachograph card in replacement of an existing card, which has been declared **as** lost, stolen or malfunctioning and has not been returned to the issuing authority. ~~Replacement always implies a risk that two valid cards may co-exist;~~

(qq) “security certification” means:

process to certify, by a Common Criteria certification body, that the ~~recording equipment~~ **control device** (or component) or the tachograph card under investigation fulfils the security requirements defined in the relative Protection Profiles;

(rr) “self test” means:

tests run cyclically and automatically by the ~~recording equipment~~ **control device** to detect faults;

(ss) “time measurement” means:

a permanent digital record of the coordinated universal date and time (UTC);

(tt) “time adjustment” means:

And ~~automatic~~ adjustment of current time; **this adjustment can be automatic at regular intervals, and within using the time provided by the GNSS receiver as a maximum tolerance of one minute reference**, or an adjustment performed ~~during-in~~ **calibration mode**;

(uu) “tyre size” means:

the designation of the dimensions of the tyres (external driving wheels) in accordance with ~~Directive 92/23/EEC of 31 march 1992⁵ as last amended~~ **UNECE Regulation 54**;

(vv) “vehicle identification” means:

numbers identifying the vehicle: Vehicle Registration Number (VRN) with indication of the registering ~~Member State~~ **Contracting Party** and Vehicle Identification Number (VIN)⁶;

(ww) for computing sake in the ~~recording equipment~~ **control device** “week” means:

the period between 00.00 hours UTC on Monday and 24.00 UTC on Sunday;

(xx) “workshop card” means:

a tachograph card issued by the authorities of a ~~Member State~~ **Contracting Party** to designated staff of a tachograph manufacturer, a fitter, a vehicle manufacturer or a workshop,

⁵—OJ No L 129, 14/05/1992, p. 0095.

approved by that ~~Member State~~ **Contracting Party**, which identifies the cardholder and allows for the testing, calibration and activation of tachographs, and/or downloading from them;

(yy) “adaptor” means:

a device, providing a signal permanently representative of vehicle speed and/or distance travelled, other than the one used for the independent movement detection, and which is:

- installed and used only in M1 and N1 type vehicles (as defined in ~~Annex II to Council Directive 2007/46/EC~~, as last amended) **put into service since 1 May 2006 Consolidated Resolution on the Construction of Vehicles (R.E.3), Revision 6, ECE/TRANS/WP.29/78/Rev.6 of 11 July 2017**);
- installed where it is not mechanically possible to install any other type of existing motion sensor which is otherwise compliant with the provisions of this ~~Annex Appendix~~ and its ~~Appendices~~ **sub-appendices 1 to 15**;
- installed between the vehicle unit and where the speed/distance impulses are generated by integrated sensors or alternative interfaces;
- seen from a vehicle unit, the adaptor behaviour is the same as if a motion sensor, compliant with the provisions of this ~~Annex Appendix~~ and its ~~Appendices~~ **sub-appendices 1 to 16**, was connected to the vehicle unit;

use of such an adaptor in those vehicles described above shall allow for the installation and correct use of a vehicle unit compliant with all the requirements of this ~~Annex Appendix~~; for those vehicles, the smart tachograph includes cables, an adaptor, and a vehicle unit;

(zz) “data integrity” means:

the accuracy and consistency of stored data, indicated by an absence of any alteration in data between two updates of a data record. Integrity implies that the data is an exact copy of the original version, e.g. that it has not been corrupted in the process of being written to, and read back from, a tachograph card or a dedicated equipment or during transmission via any communications channel;

~~(aaa) data privacy means:~~

~~the overall technical measures taken to ensure the proper implementation of the principles laid down in Directive 95/46/EC of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data as well as of those laid down in Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector;~~

(aaa) reserved

(bbb) “smart tachograph system” means:

~~the recording equipment control device~~, tachograph cards and the set of all directly or indirectly interacting equipment during their construction, installation, use, testing and control, such as cards, remote communication reader and any other equipment for data downloading, data analysis, calibration, generating, managing or introducing security elements, etc;

Smart tachographs are control devices complying with Appendix 1C of this Agreement

(ccc) “introduction date” means:

~~(ccc) 36 months after the entry into force of the detailed provisions referred to in Article 11 of Regulation (EU) N°. 165/2014. This is the~~

~~The~~ **the date after which vehicles registered for the first time shall be fitted with a tachograph in accordance with this Agreement;-**

~~shall be fitted with a tachograph connected to a positioning service based on a satellite navigation system,~~

~~shall be able to communicate data for targeted roadside checks to competent control authorities while the vehicle is in motion,~~

~~and may be equipped with standardised interfaces allowing the data recorded or produced by tachographs to be used in operational mode, by an external device.~~

(ddd) “protection profile” means:

a document used as part of certification process according Common Criteria, providing implementation independent specification of information assurance security requirements;

(eee) “GNSS accuracy” means:

in the context of recording the position from Global Navigation Satellite System (GNSS) with tachographs, means the value of the Horizontal Dilution of Precision (HDOP) calculated as the minimum of the HDOP values collected on the available GNSS systems.

(fff) “accumulated driving time” means:

a value representing the total accumulated number of minutes of driving of a particular vehicle.

The accumulated driving time value is a free running count of all minutes regarded as DRIVING by the monitoring of driving activities function of the control device, and is only used for triggering the recording of the vehicle position, every time a multiple of three hours of accumulated driving is reached. The accumulation is started at the control device activation. It is not affected by any other condition, like out of scope or ferry/train crossing. The accumulated driving time is not intended to be displayed, printed or downloaded.

(ggg) “Mass of the unladen vehicle in running order” means:

(a) in the case of a motor vehicle:

the mass of the vehicle, with its fuel tank(s) filled to at least 90 % of its or their capacity/ies, including the mass of the driver, of the fuel and liquids, fitted with the standard equipment in accordance with the manufacturer’s specifications and, when they are fitted, the mass of the bodywork, the cabin, the coupling and the spare wheel(s) as well as the tools;

(b) in the case of a trailer:

the mass of the vehicle including the fuel and liquids, fitted with the standard equipment in accordance with the manufacturer’s specifications, and, when they are fitted, the mass of the bodywork, additional coupling(s), the spare wheel(s) and the tools;

(hhh) “Vehicle Identification Number” means:

a fixed combination of characters assigned to each vehicle by the manufacturer, which consists of two sections: the first, composed of not more than six characters (letters or figures), identifying the general characteristics of the vehicle, in particular the type and model; the second, composed of eight characters of which the first four may be letters or figures and the other four figures only, providing, in conjunction with the first section, clear identification of a particular vehicle.

2. General characteristics and functions of the recording equipment control device

2.1. General characteristics

The purpose of the ~~recording equipment~~ **control device** is to record, store, display, print, and output data related to driver activities.

Any vehicle fitted with the ~~recording equipment~~ **control device** complying with the provisions of this ~~Annex~~ **Appendix**, must include a speed display and an odometer. These functions may be included within the ~~recording equipment~~ **control device**.

(1) The ~~recording equipment~~ **control device** includes cables, a motion sensor, and a vehicle unit

(2) The interface between motion sensors and vehicle units shall comply with the requirements specified in ~~Appendix~~ **Sub-appendix 11**.

(3) The vehicle unit shall be connected to global navigation satellite system(s), as specified in ~~Appendix~~ **Sub-appendix 12**.

(4) The vehicle unit shall communicate with remote early detection communication readers, as specified in ~~Appendix~~ **Sub-appendix 14**.

(5) The vehicle unit ~~may~~ **shall** include an ITS interface, which is specified in ~~Appendix~~ **Sub-appendix 13**. The ~~recording equipment~~ **control device** may be connected to other facilities through additional interfaces and/or through the ~~optional~~ ITS interface.

(6) Any inclusion in or connection to the ~~recording equipment~~ **control device** of any function, device, or devices, approved or otherwise, shall not interfere with, or be capable of interfering with, the proper and secure operation of the ~~recording equipment~~ **control device** and the provisions ~~Regulation~~ of this **Agreement**. ~~Recording equipment~~ **Control device** users identify themselves to the equipment via tachograph cards.

(7) The ~~recording equipment~~ **control device** provides selective access rights to data and functions according to user's type and/or identity. The ~~recording equipment~~ **control device** records and stores data in its data memory, in the remote communication facility and in tachograph cards.

~~This is done in accordance with Directive 95/46/EC of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data⁷, with Directive 2002/58/EC of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector⁸ and in compliance with Article 7 of Regulation (EU) N° 165/2014.~~

2.2. Functions

(8) The ~~recording equipment~~ **control device** shall ensure the following functions:

- monitoring cards insertions and withdrawals,
- speed, distance and position measurement,
- time measurement,

⁷—OJ No L 281, 23/11/1995, p.31.

⁸—OJ N° L 201, 31/07/2002, p.37

- monitoring driver activities,
- monitoring driving status,
- drivers manual entries:
 - entry of places where daily work periods begin and/or end,
 - manual entry of driver activities **and driver consent for ITS interface**,
 - entry of specific conditions,
 - **entry of load or unload operations**,
- company locks management,
- monitoring control activities,
- detection of events and/or faults,
- built-in and self-tests,
- reading from data memory,
- recording and storing in data memory,
- reading from tachograph cards,
- recording and storing in tachograph cards,
- displaying,
- printing,
- warning,
- data downloading to external media,
- remote communication for targeted roadside checks,
- output data to additional facilities,
- calibration,
- roadside calibration check,
- time adjustment,
- **monitoring border crossings**,
- **software update**.

2.3. Modes of operation

- (9) The ~~recording equipment~~ **control device** shall possess four modes of operation:
- operational mode,
 - control mode,
 - calibration mode,
 - company mode.
- (10) The ~~recording equipment~~ **control device** shall switch to the following mode of operation according to the valid tachograph cards inserted into the card interface devices. In

order to determine the mode of operation, the tachograph card generation is irrelevant, provided the inserted card is valid. A first generation workshop card shall always be considered as non-valid when it is inserted in a second generation VU.

Mode of operation		Driver slot				
		No card	Driver card	Control card	Workshop card	Company card
Co-driver slot	No card	Operational	Operational	Control	Calibration	Company
	Driver card	Operational	Operational	Control	Calibration	Company
	Control card	Control	Control	Control (*)	Operational	Operational
	Workshop card	Calibration	Calibration	Operational	Calibration (*)	Operational
	Company card	Company	Company	Operational	Operational	Company (*)

(*) In these situations the ~~recording equipment~~ **control device** shall use only the tachograph card inserted in the driver slot.

(11) The ~~recording equipment~~ **control device** shall ignore non valid cards inserted, except displaying, printing or downloading data held on an expired card which shall be possible.

(12) All functions listed in 2.2. shall work in any mode of operation with the following exceptions:

- the calibration function is accessible in the calibration mode only,
- the roadside calibration checking function is accessible in the control mode only,
- the company locks management function is accessible in the company mode only,
- the monitoring of control activities function is operational in the control mode only,
- The downloading function is not accessible in the operational mode, (except
 - as provided for in requirement 193),
 - ~~and except~~ downloading a driver card when no other card type is inserted into the VU.

(13) The ~~recording equipment~~ **control device** can output any data to display, printer or external interfaces with the following exceptions:

- in the operational mode, any personal identification (surname and first name(s)) not corresponding to a tachograph card inserted shall be blanked and any card number not corresponding to a tachograph card inserted shall be partially blanked (every odd character – from left to right - shall be blanked),
- in the company mode, driver related data (requirements 102, 105, ~~and 108, 133a and 133e~~) can be output only for periods where no lock exists or no other company holds a lock (as identified by the first 13 digits of the company card number),
- when no card is inserted in the ~~recording equipment~~ **control device**, driver related data can be output only for the current and 8 previous calendar days,
- personal data **recorded and produced by either the tachograph or the tachograph cards** ~~originating from the VU~~ shall not be output through ITS interface of the VU unless the consent of the driver to whom the data relates is verified,
- the vehicle units have a normal operations validity period of 15 years, starting with the vehicle unit certificates ~~issuing~~ **effective** date, but vehicle units can be used for additional 3 months, for data downloading only.

2.4. Security

The system security aims at protecting the data memory in such a way as to prevent unauthorised access to and manipulation of the data and detecting any such attempts, protecting the integrity and authenticity of data exchanged between the motion sensor and the vehicle unit, protecting the integrity and authenticity of data exchanged between the ~~recording equipment~~ **control device** and the tachograph cards, protecting the integrity and authenticity of data exchanged between the ~~recording equipment~~ **control device** and the external GNSS facility, **if any**, protecting the confidentiality, integrity and authenticity of data exchanged through the remote early detection communication for control purposes, and verifying the integrity and authenticity of data downloaded.

(14) In order to achieve the system security, the following components shall meet the security requirements specified in their Protection Profiles, as required in ~~Appendix~~ **sub-appendix 10**:

- vehicle unit,
- tachograph card,
- motion sensor,
- external GNSS facility (this Profile is only needed and applicable for the external GNSS **facility** variant).

3. Construction and functional requirements for ~~recording equipment~~ **control device**

3.1. Monitoring cards ~~insertion~~ and withdrawal

(15) The ~~recording equipment~~ **control device** shall monitor the card interface devices to detect card insertions and withdrawals.

(16) Upon card insertion (**or remote card authentication**) the ~~recording equipment~~ **control device** shall detect whether the card ~~inserted~~ is a valid tachograph card **in accordance with definition (ee) in section 1**, and in such a case identify the card type and the card generation.

~~If a card with the same card number and a higher renewal index has already been inserted in the recording equipment control device, the card shall be declared as non valid. If a card with the same card number and renewal index but with a higher replacement index has already been inserted in the recording equipment control device, the card shall be declared as non valid~~

For checking if a card has already been inserted, the recording equipment shall use the tachograph card data stored in its data memory, as set out in requirement 133.

(17) First generation tachograph cards shall be considered as non-valid by the ~~recording equipment~~ **control device**, after the possibility of using first generation tachograph cards has been suppressed by a workshop, in compliance with ~~Appendix~~ **sub-appendix 15** (req. ~~MIG003~~ **MIG_003**).

(18) First generation workshop cards which are inserted in the second generation ~~recording equipment~~ **control device** shall be considered as non-valid.

(19) The ~~recording equipment~~ **control device** shall be so designed that the tachograph cards are locked in position on their proper insertion into the card interface devices.

(20) The ~~withdrawal~~~~release~~ of tachograph cards may function only when the vehicle is stopped and after the relevant data have been stored on the cards. The ~~release~~~~withdrawal~~ of the card shall require positive action by the user.

3.2. Speed, position and distance measurement

(21) The motion sensor (possibly embedded in the adaptor) is the main source for speed and distance measurement.

(22) This function shall continuously measure and be able to provide the odometer value corresponding to the total distance travelled by the vehicle using the pulses provided by the motion sensor.

(23) This function shall continuously measure and be able to provide the speed of the vehicle using the pulses provided by the motion sensor.

(24) The speed measurement function shall also provide the information whether the vehicle is moving or stopped. The vehicle shall be considered as moving as soon as the function detects more than 1 imp/sec for at least 5 seconds from the motion sensor, otherwise the vehicle shall be considered as stopped.

(25) Devices displaying speed (speedometer) and total distance travelled (odometer) installed in any vehicle fitted with a ~~recording equipment~~ **control device** complying with the provisions of this ~~Regulation Agreement~~, shall comply with the requirements relating to maximum tolerances (see 3.2.1 and 3.2.2) laid down in this ~~Annex~~ **Appendix**.

(26) To detect manipulation of motion data, information from the motion sensor shall be corroborated by vehicle motion information derived from the GNSS receiver and ~~by optionally by~~ other source(s) independent from the motion sensor. **At least another independent vehicle motion source shall be inside the VU without the need of an external interface.**

(27) This function shall measure the position of the vehicle in order to allow for the ~~automatic~~ recording of:

- positions where the driver and/or the co-driver begins his daily work period;
- positions where the ~~continuous~~ **accumulated** driving time of the driver reaches a multiple of three hours;
- **positions where the vehicle has crossed the border of a country;**
- **positions where operations of load or unload have been carried out;**
- positions where the driver and/or the co-driver ends his daily work period.

3.2.1 Measurement of distance travelled

(28) The distance travelled may be measured either:

- so as to cumulate both forward and reverse movements, or
- so as to include only forward movement.

(29) The ~~recording equipment~~ **control device** shall measure distance from 0 to 9 999 999.9 km.\$

(30) Distance measured shall be within the following tolerances (distances of at least 1000 m.):

- $\pm 1\%$ before installation,

- $\pm 2\%$ on installation and periodic inspection,
- $\pm 4\%$ in use.

The tolerances shall not be used to intentionally alter the distance measured.

(31) Distance measured shall have a resolution better than or equal to 0.1 km.

3.2.2 Measurement of speed

(32) The ~~recording equipment~~ **control device** shall measure speed from 0 to 220 km/h.

(33) To ensure a maximum tolerance on speed displayed of ± 6 km/h in use, and taking into account:

- a ± 2 km/h tolerance for input variations (tyre variations, ...),
- a ± 1 km/h tolerance in measurements made during installation or periodic inspections,

the ~~recording equipment~~ **control device** shall, for speeds between 20 and 180 km/h, and for characteristic coefficients of the vehicle between 24000 and 25000 imp/km, measure the speed with a tolerance of ± 1 km/h (at constant speed).

Note: The resolution of data storage brings an additional tolerance of ± 0.5 km/h to speed stored by the ~~recording equipment~~ **control device**.

(34) The speed shall be measured correctly within the normal tolerances within 2 seconds of the end of a speed change when the speed has changed at a rate up to 2m/s².

(35) Speed measurement shall have a resolution better than or equal to 1 km/h.

3.2.3 Measurement of position

(36) The ~~recording equipment~~ **control device** shall measure the absolute position of the vehicle using the GNSS receiver.

(37) The absolute position **shall be** measured in geographical coordinates of latitude and longitude in degrees and minutes with a resolution of 1/10 of a minute.

3.3. Time measurement

(38) The time measurement function shall measure permanently and digitally provide UTC date and time.

(39) UTC date and time shall be used for dating data inside the ~~recording equipment~~ **control device** (recordings, data exchange) and for all printouts specified in ~~Appendix~~ **sub-appendix 4** "Printouts".

(40) In order to visualise the local time, it shall be possible to change the offset of the time displayed, in half hour steps. No other offsets than negative or positive multiples of half hours shall be allowed;

(41) Time drift shall be within ± 21 seconds per day **or less**, in ~~temperature type approval~~ conditions **in accordance with requirement 213**, in the absence of any time adjustment.

(41a) Time accuracy when time is adjusted by workshops in accordance with requirement 212 shall be 3 seconds or better.

(41b) The vehicle unit shall manage a drift counter, which computes the maximal time drift since the last time adjustment in accordance with point 3.23. The maximal time drift shall be defined by the vehicle unit manufacturer and shall not exceed 1 second per day, as set out in requirement 41.

(41c) The drift counter shall be reset to 1 second after each time adjustment of the recording equipment in accordance with point 3.23. This includes:

- **automatic time adjustments,**
- **time adjustments performed in calibration mode.**

(42) Time measured shall have a resolution better than or equal to 1 second.

(43) Time measurement shall not be affected by an external power supply cut-off of less than 12 months in type approval conditions.

3.4. Monitoring driver activities

(44) This function shall permanently and separately monitor the activities of one driver and one co-driver.

(45) Driver activity shall be DRIVING, WORK, AVAILABILITY or BREAK/REST.

(46) It shall be possible for the driver and/or the co-driver to manually select WORK, AVAILABILITY or BREAK/REST.

(47) When the vehicle is moving, DRIVING shall be selected automatically for the driver and AVAILABILITY shall be selected automatically for the co-driver.

(48) When the vehicle stops, WORK shall be selected automatically for the driver.

(49) The first change of activity to **BREAK/REST** or AVAILABILITY arising within 120 seconds of the automatic change to WORK due to the vehicle stop shall be assumed to have happened at the time of vehicle stop (therefore possibly cancelling the change to WORK)

(50) This function shall output activity changes to the recording functions at a resolution of one minute.

(51) Given a calendar minute, if DRIVING is registered as the activity of both the immediately preceding and the immediately succeeding minute, the whole minute shall be regarded as DRIVING.

(52) Given a calendar minute that is not regarded as DRIVING according to requirement 051, the whole minute shall be regarded to be of the same type of activity as the longest continuous activity within the minute (or the latest of the equally long activities).

(53) This function shall also permanently monitor the continuous driving time and the cumulative break time of the driver.

3.5. Monitoring driving status

(54) This function shall permanently and automatically monitor the driving status.

(55) The driving status CREW shall be selected when two valid driver cards are inserted in the equipment, the driving status SINGLE shall be selected in any other case.

3.6. Drivers entries

3.6.1 Entry of places where daily work periods begin and/or end

(56) This function shall allow for the entry of places where, according to the driver and/or the co-driver, his daily work periods begin and/or end.

(57) Places are defined as the country and, in addition where applicable, the region, ~~which are entered or confirmed manually.~~

(58) ~~Upon~~**At the time of a driver (or workshop) card withdrawal, the recording equipment control device shall display the current place of the vehicle on the basis of the GNSS information, and of stored digital map in accordance with point 3.12.19, and shall request the cardholder to confirm or to manually rectify the place** ~~prompt the (co-)driver to enter a "place where the daily work period ends".~~

(59) **The place entered in accordance with requirement 58 shall be considered as the place where the daily work period ends. It shall be recorded in the relevant driver (or workshop) card as a temporary record, and may therefore be later overwritten**~~The driver shall then enter the current place of the vehicle, which shall be considered as a temporary entry.~~

Under the following conditions temporary entry made at last card withdrawal is validated (i.e. shall not be overwritten anymore):

- entry of a place where the current daily work period begins during manual entry according to requirement (61);
- the next entry of a place where the current daily work period begins if the card holder does ~~not~~**not** enter any place where the work period begins or ended during the manual entry according to requirement (61).

Under the following conditions temporary entry made at last card withdrawal is overwritten and the new value is validated:

- the next entry of a place where the current daily work period ends if the card holder does not enter any place where the work period begins or ended during the manual input according to requirement (61).

(60) It shall be possible to input places where daily work periods begin and/or end through commands in the menus. If more than one such input is done within one calendar minute, only the last begin place input and the last end place input done within that time shall be kept recorded.

The recording equipment shall display the current place of the vehicle on the basis of the GNSS information, and of stored digital map in accordance with point 3.12.19 and shall request the driver to confirm or to manually rectify the place.

3.6.2 Manual entry of driver activities and driver consent for ITS interface

(61) Upon driver (or workshop) card insertion, and only at this time, the ~~recording equipment control device~~ **recording equipment control device** shall allow manual entries of activities. Manual entries of activities shall be performed using local time and date values of the time zone (UTC offset) currently set for the vehicle unit.

At driver or workshop card insertion the cardholder shall be reminded of:

- the date and time of his last card withdrawal;
- optionally: the local time offset currently set for the vehicle unit.

At the first insertion of a given driver card or workshop card currently unknown to the vehicle unit, the cardholder shall be invited to express his consent for tachograph related personal data output through the ~~optional~~ **optional** ITS interface. **For checking if a card has already been inserted, the recording equipment shall use the tachograph card data stored in its data memory, as set out in requirement 133.**

At any moment, the driver (resp. workshop) consent can be enabled or disabled through commands in the menu, provided the driver (resp. workshop) card is inserted.

It shall be possible to input activities with the following restrictions:

- Activity type shall be WORK, AVAILABILITY or BREAK/REST;
- Start and end times for each activity shall be within the period of the last card withdrawal – current insertion only;
- Activities shall not be allowed to overlap mutually in time.

It shall be possible to make manual entries, if required, at the first insertion of a previously unused driver (or workshop) card.

The procedure for manual entries of activities shall include as many consecutive steps as necessary to set a type, a start time and an end time for each activity. For any part of the time period between last card withdrawal and current card insertion, the cardholder shall have the option not to declare any activity.

During the manual entries associated with card insertion and if applicable, the card holder shall have the opportunity to input:

- a place where a previous daily work period ended, associated to the relevant time (thus overwriting **and validating** the entry made at the last card withdrawal),
- a place where the current daily work period begins, associated to the relevant time (**thus validating a temporary entry made at last card withdrawal**).

For the place where the current daily work period begins entered at the current card insertion, the recording equipment shall display the current place of the vehicle on the basis of the GNSS information, and of stored digital map(s) in accordance with point 3.12.19, and shall request the driver to confirm or to manually rectify the place.

If the card holder does **not** enter any place where the work period begins or ended, during the manual entries associated with card insertion, this shall be considered as a declaration that his work period has not changed since the last card withdrawal. The next entry of a place where a previous daily work period ends shall then overwrite the temporary entry made at the last card withdrawal.

If a place is entered, it shall be recorded in the relevant tachograph card.

Manual entries shall be interrupted if:

- the card is withdrawn or,
- the vehicle is moving and the card is in the driver slot.

Additional interruptions are allowed, e.g. a timeout after a certain period of user inactivity. If manual entries are interrupted, the ~~recording equipment~~ **control device** shall validate any complete place and activity entries (having either unambiguous place and time, or activity type, begin time and end time) already made.

If a second driver or workshop card is inserted while manual entries of activities are in progress for a previously inserted card, the manual entries for this previous card shall be allowed to be completed before manual entries start for the second card.

The cardholder shall have the option to insert manual entries according to the following minimum procedure:

- Enter activities manually, in chronological order, for the period last card withdrawal – current insertion.

- Begin time of the first activity shall be set to card withdrawal time. For each subsequent entry, the start time shall be preset to immediately follow the end time of the previous entry. Activity type and end time shall be selected for each activity.

The procedure shall end when the end time of a manually entered activity equals the card insertion time.

The recording equipment shall allow drivers and workshops to alternately upload manual entries that need to be entered during the procedure through the ITS interface specified in Sub-appendix 13 and, optionally, through other interfaces.

The ~~recording equipment control device may~~ **shall** ~~then optionally~~ allow the card holder to modify any activity manually entered, until validation by selection of a specific command. Thereafter, any such modification shall be forbidden.

3.6.3 Entry of specific conditions

(62) The ~~recording equipment control device~~ shall allow the driver to enter, in real time, the following two specific conditions:

- “OUT OF SCOPE” (begin, end)
- “FERRY / TRAIN CROSSING” (begin, end).

A “FERRY / TRAIN CROSSING” ~~may~~ **shall** not occur if an “OUT OF SCOPE” condition is opened. **If an “OUT OF SCOPE” condition is opened, the recording equipment shall not allow users to enter a “FERRY / TRAIN CROSSING” begin flag.**

An opened “OUT OF SCOPE” condition must be automatically closed, by the ~~recording equipment control device~~, if a driver card is inserted or withdrawn.

An opened "OUT OF SCOPE" condition shall inhibit the following events and warnings:

- Driving without an appropriate card,
- Warnings associated with continuous driving time.

The **driver shall enter the FERRY / TRAIN CROSSING begin flag immediately after selecting BREAK/REST** ~~shall be set before shutting down the engine~~ on the ferry/train.

An opened FERRY / TRAIN CROSSING must **be ended by the control device** when any of ~~the~~ following options occurs:

- ~~¶~~the driver manually ends the FERRY/TRAIN CROSSING, **which shall occur upon arrival to destination of the ferry/train, before driving off the ferry/train,**
- **an "OUT OF SCOPE" condition is opened,**
- ~~¶~~the driver ejects his card,
- **driver activity is computed as DRIVING during a calendar minute in accordance with point 3.4.**

If more than one specific conditions entry of the same type is done within one calendar minute, only the last one shall be kept recorded.

~~An opened FERRY/TRAIN CROSSING shall end when it is no longer valid based on the rules stated in Regulation (EC) N°. 561/2006 this Agreement.~~

3.6.4 Entry of load/unload operation

(62a) The recording equipment shall allow the driver to enter and confirm, in real time, information *indicating* that the vehicle is being loaded, unloaded or that simultaneous load/unload operation is being performed.

If more than one load/unload operation entry of the same type is done within one calendar minute, only the last one shall be kept recorded.

(62b) Load, unload or simultaneous load/unload operations shall be recorded as separate events.

(62c) The load/unload information shall be entered before the vehicle leaves the place where the load/unload operation is carried out.

3.7. Company locks management

(63) This function shall allow the management of the locks placed by a company to restrict data access in company mode to itself.

(64) Company locks consist in a start date/time (lock-in) and an end date/time (lock-out) associated with the identification of the company as denoted by the company card number (at lock-in).

(65) Locks may be turned “in” or “out” in real time only.

(66) Locking-out shall only be possible for the company whose lock is “in” (as identified by the first 13 digits of the company card number), or,

(67) Locking-out shall be automatic if another company locks in.

(68) In the case where a company locks in and where the previous lock was for the same company, then it will be assumed that the previous lock has not been turned “out” and is still “in”.

3.8. Monitoring control activities

(69) This function shall monitor DISPLAYING, PRINTING, VU and card DOWNLOADING, and ROADSIDE CALIBRATION check activities carried while in control mode.

(70) This function shall also monitor OVER SPEEDING CONTROL activities while in control mode. An over speeding control is deemed to have happened when, in control mode, the “over speeding” printout has been sent to the printer or to the display, or when “events and faults” data have been downloaded from the VU data memory.

3.9. Detection of events and/or faults

(71) This function shall detect the following events and/or faults:

3.9.1 “Insertion of a non-valid card” event

(72) This event shall be triggered at the insertion of any non-valid card, at the insertion of a driver card already replaced and/or when an inserted valid card expires.

3.9.2 “Card conflict” event

(73) This event shall be triggered when any of the valid cards combination noted X in the following table arises:

Card conflict		Driver slot				
		No card	Driver card	Control card	Workshop card	Company card
Co-driver slot	No card					
	Driver card				X	
	Control card			X	X	X
	Workshop card		X	X	X	X
	Company card			X	X	X

3.9.3 “Time overlap” event

(74) This event shall be triggered when the date / time of last withdrawal of a driver card, as read from the card, is later than the current date / time of the ~~recording equipment~~ **control device** in which the card is inserted.

3.9.4 “Driving without an appropriate card” event

(75) This event shall be triggered for any valid tachograph cards combination noted X in the following table, when driver activity changes to DRIVING, or when there is a change of the mode of operation while driver activity is DRIVING:

Driving without an appropriate card		Driver slot				
		No (or non-valid) card	Driver card	Control card	Workshop card	Company card
Co-driver slot	No (or non-valid) card	X		X		X
	Driver card	X		X	X	X
	Control card	X	X	X	X	X
	Workshop card	X	X	X		X
	Company card	X	X	X	X	X

3.9.5 “Card insertion while driving” event

(76) This event shall be triggered when a tachograph card is inserted in any slot, while driver activity is DRIVING.

3.9.6 “Last card session not correctly closed” event

(77) This event shall be triggered when at card insertion the ~~recording equipment~~ **control device** detects that, despite the provisions laid down in paragraph 3.1, the previous card session has not been correctly closed (the card has been withdrawn before all relevant data have been stored on the card). This event shall be triggered by driver and workshop cards only.

3.9.7 “Over speeding” event

(78) This event shall be triggered for each over speeding.

3.9.8 “Power supply interruption” event

(79) This event shall be triggered, while not in calibration or control mode, in case of any interruption exceeding 200 milliseconds of the power supply of the motion sensor and/or of the vehicle unit. The interruption threshold shall be defined by the manufacturer. The drop in power supply due to the starting of the engine of the vehicle shall not trigger this event.

3.9.9 “Communication error with the remote communication facility” event

(80) This event shall be triggered, while not in calibration mode, when the remote communication facility does not acknowledge the successful reception of remote communication data sent from the vehicle unit for more than three attempts.

3.9.10 “Absence of position information from GNSS receiver” event

(81) This event shall be triggered, while not in calibration mode, in case of absence of position information originating from the GNSS receiver (whether internal or external) for more than three hours of accumulated driving time.

3.9.11 “Communication error with the external GNSS facility” event

(82) This event shall be triggered, while not in calibration mode, in case of interruption of the communication between the external GNSS facility and the vehicle unit for more than 20 continuous minutes, when the vehicle is moving.

3.9.12 “Motion data error” event

(83) This event shall be triggered, while not in calibration mode, in case of interruption of the normal data flow between the motion sensor and the vehicle unit and/or in case of data integrity or data authentication error during data exchange between the motion sensor and the vehicle unit. **This event shall also be triggered, while not in calibration mode, in case the speed calculated from the motion sensor pulses increases from 0 to more than 40 km/h within 1 second, and then stays above 40km/h during at least 3 seconds.**

3.9.13 “Vehicle motion conflict” event

(84) This event shall be triggered, **as specified in Sub-appendix 12**, while not in calibration mode, in case motion information calculated from the motion sensor is contradicted by motion information calculated from the internal GNSS receiver or from the external GNSS facility ~~or and optionally~~ by other independent source(s) **in accordance with requirement 26, as specified in Sub-Appendix 12**. This event shall not be triggered during a ferry/train crossing, an OUT-OF-SCOPE condition, or when the position information from the GNSS receiver is not available.

3.9.14 “Security breach attempt” event

(85) This event shall be triggered for any other event affecting the security of the motion sensor and/or of the vehicle unit and/or the external GNSS facility as required in ~~Appendix~~ **Sub-appendix 10**, while not in calibration mode.

3.9.15 “Time conflict” event

(86) This event shall be triggered, while not in calibration mode, when the VU detects a discrepancy ~~of more than 1 minute~~ between the time of the vehicle unit’s time measurement function and the time originating from the **authenticated positions transmitted by the GNSS receiver or the external GNSS facility**. A **“time discrepancy” is detected if the time difference exceeds ± 3 seconds corresponding to the time accuracy set out in requirement 41a, the latter increased by the maximal time drift per day.** This event is

~~shall be recorded together with the internal clock value of the vehicle control device, and comes together with an automatic time adjustment. After a time conflict event has been triggered, the VU will not generate other time conflict events for the next 12 hours. This event shall not be triggered in cases where no valid GNSS signal was detectable by the GNSS receiver within the last for 30 days. The VU shall perform the check for triggering the “time conflict” event right before the VU automatically re-adjusts the VU internal clock, in accordance with requirement 211. However, when the position information from the GNSS receiver is available again, the automatic time adjustment shall be done. or more.~~

3.9.16 “Card” fault

(87) This fault shall be triggered when a tachograph card failure occurs during operation.

3.9.17 “~~Recording equipment~~ Control device” fault

(88) This fault shall be triggered for any of these failures, while not in calibration mode:

- VU internal fault
- Printer fault
- Display fault
- Downloading fault
- Sensor fault
- GNSS receiver or external GNSS facility fault
- Remote Communication facility fault
- **ITS interface fault (if applicable).**

3.9.18 “GNSS anomaly” event

(88a) **This event shall be triggered, while not in calibration mode, when the GNSS receiver detects an attack, or when authentication of navigation messages has failed, as specified in Sub-appendix 12. After a GNSS anomaly event has been triggered, the VU shall not generate other GNSS anomaly events for the next 10 minutes.**

3.10. “Built-in and self-tests”

(89) ~~The recording equipment control device shall self~~ detect faults through self-tests and built-in-tests, according to the following table:

<i>Sub-assembly to test</i>	<i>self-test</i>	<i>Built-in-test</i>
Software		Integrity
Data memory	Access	Access, data integrity
Card interface devices	Access	Access
Keyboard		Manual check
Printer	(up to manufacturer)	Printout
Display		Visual check
Downloading (performed only during downloading)	Proper operation	
Sensor	Proper operation	Proper operation
Remote communication facility	Proper operation	Proper operation
GNSS facility	Proper operation	Proper operation
ITS interface (optional)	Proper operation	

3.11. Reading from data memory

(90) The ~~recording equipment~~ **control device** shall be able to read any data stored in its data memory.

3.12. Recording and storing in the data memory

For the purpose of this paragraph,

- “365 days” is defined as 365 calendar days of average drivers’ activity in a vehicle. The average activity per day in a vehicle is defined as at least 6 drivers or co-drivers, 6 card insertion withdrawal cycles, and 256 activity changes. “365 days” therefore include at least 2190 (co-)drivers, 2190 card insertion withdrawal cycles, and 93440 activity changes,
- **the average number of place entries per day is defined as at least 6 entries where the daily work period begins and 6 positions where the daily work period ends, so that “365 days” include at least 4380 place entries,**
- the average number of positions per day **when the accumulated driving time reaches a multiple of three hours is defined as at least 6 positions, so that “365” days include at least 2190 such** ~~where the daily work period begins, 6 positions when the driver’s continuous accumulated driving time reaches a multiple of three hours, and 6 positions where the daily work period ends, so that “365 days” include at least 6570 positions,~~
- **the average number of border crossings per day is defined as at least 20 crossings, so that “365 days” include at least 7300 border crossings,**
- **the average number of load/unload operations per day is defined as at least 25 operations (irrespective of the type), so that “365 days” include at least 9125 load/unload operations,**
- times are recorded with a resolution of one minute, unless otherwise specified,
- odometer values are recorded with a resolution of one kilometre,
- speeds are recorded with a resolution of 1 km/h,
- positions (latitudes and longitudes) are recorded in degrees and minutes, with a resolution of 1/10 of minute, with the associated GNSS accuracy and acquisition time, **and with a flag indicating whether the position has been authenticated.**

(91) Data stored into the data memory shall not be affected by an external power supply cut-off of less than twelve months in type approval conditions. In addition, data stored in the external remote communication facility, as defined in ~~Appendix~~ **Sub-appendix** 14, shall not be affected by power-supply cut-off of less than 28 days.

(92) The ~~recording equipment~~ **control device** shall be able to record and store implicitly or explicitly in its data memory the following:

3.12.1 Equipment identification data

3.12.1.1 Vehicle Unit identification data

(93) The ~~recording equipment~~ **control device** shall be able to store in its data memory the following vehicle unit identification data:

name of the manufacturer,

- address of the manufacturer,
- part number,
- serial number,
- VU generation,
- ability to use first generation tachograph cards,
- software version number,
- software version installation date,
- year of equipment manufacture,
- approval number,
- **digital map version identifier (requirement 133l).**

(94) Vehicle unit identification data are recorded and stored once and for all by the vehicle unit manufacturer, except ~~data the software related data and the approval number~~ which may be changed in case of software update/grade and the ability to use first generation tachograph cards.

3.12.1.2 Motion sensor identification data

(95) The motion sensor shall be able to store in its memory the following identification data:

- name of the manufacturer,
- serial number,
- approval number,
- embedded security component identifier (e.g. internal chip/processor part number),
- operating system identifier (e.g. software version number).

(96) Motion sensor identification data are recorded and stored once and for all in the motion sensor, by the motion sensor manufacturer.

(97) The vehicle unit shall be able to record and store in its data memory the following data related to the 20 most recent **successful** pairings of motion sensors (if several pairings happen within one calendar day, only the first and the last one of the day shall be stored):

The following data shall be recorded for each of these pairings:

- motion sensor identification data:
 - serial number
 - approval number
- motion sensor pairing data:
 - pairing date.

3.12.1.3 Global Navigation Satellite Systems identification data

(98) The external GNSS facility shall be able to store in its memory the following identification data:

- name of the manufacturer,

- serial number,
- approval number,
- embedded security component identifier (e.g. internal chip/processor part number),
- operating system identifier (e.g. software version number).

(99) The identification data are recorded and stored once and for all in the external GNSS facility, by the external GNSS facility manufacturer.

(100) The vehicle unit shall be able to record and store in its data memory the following data related to the 20 most recent **successful** couplings of external GNSS facilities (if several couplings happen within one calendar day, only the first and the last one of the day shall be stored).

The following data shall be recorded for each of these couplings:

- external GNSS facility identification data:
 - serial number,
 - approval number,
- external GNSS facility coupling data:
 - coupling date

3.12.2 Keys and Certificates

(101) The ~~recording equipment~~ **control device** shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part A and part B.

3.12.3 Driver or workshop card insertion and withdrawal data

(102) For each insertion and withdrawal cycle of a driver or workshop card in the ~~recording equipment~~, **control device** shall record and store in its data memory:

- the card holder's surname and first name(s) as stored in the card,
- the card's number, issuing ~~Member State~~ **Contracting Party** and expiry date as stored in the card,
- the card generation,
- the insertion date and time,
- the vehicle odometer value at card insertion,
- the slot in which the card is inserted,
- the withdrawal date and time,
- the vehicle odometer value at card withdrawal,
- the following information about the previous vehicle used by the driver, as stored in the card:
 - VRN and registering ~~Member State~~ **Contracting Party**,
 - VU generation (when available),
 - card withdrawal date and time,

- a flag indicating whether, at card insertion, the card holder has manually entered activities or not.

(103) The data memory shall be able to hold these data for at least 365 days.

(104) When storage capacity is exhausted, new data shall replace oldest data.

3.12.4 Driver activity data

(105) The ~~recording equipment~~ **control device** shall record and store in its data memory whenever there is a change of activity for the driver and/or the co-driver, and/or whenever there is a change of driving status, and/or whenever there is an insertion or withdrawal of a driver or workshop card:

- the driving status (CREW, SINGLE),
- the slot (DRIVER, CO-DRIVER),
- the card status in the relevant slot (INSERTED, NOT INSERTED),
- the activity (DRIVING, AVAILABILITY, WORK, BREAK/REST),
- the date and time of the change.

INSERTED means that a valid driver or workshop card is inserted in the slot. NOT INSERTED means the opposite i.e. no valid driver or workshop card is inserted in the slot (e.g. a company card is inserted or no card is inserted)

Activity data manually entered by a driver are not recorded in the data memory.

(106) The data memory shall be able to hold driver activity data for at least 365 days.

(107) When storage capacity is exhausted, new data shall replace oldest data.

3.12.5 Places and positions where daily work periods begin, end, and/or where 3 hours ~~continuous~~ accumulated driving time is reached

(108) The ~~recording equipment~~ **control device** shall record and store in its data memory:

- places and positions where the driver and/or the co-driver begins his daily work period;
- positions where the ~~continuous~~ **accumulated** driving time of the driver reaches a multiple of three hours;
- places and positions where the driver and/or the co-driver ends his daily work period.

(109) When the position of the vehicle is not available from the GNSS receiver at these times, the ~~recording equipment~~ **control device** shall use the latest available position, and the related date and time.

110) Together with each place or position, ~~recording equipment~~ **the control device** shall record and store in its data memory:

- the (co-)driver card number and card issuing ~~Member State~~ **Contracting Party**,
- the card generation,
- the date and time of the entry,
- the type of entry (begin, end or 3 hours ~~continuous~~ **accumulated** driving time),
- the related GNSS accuracy, date and time if applicable,;

- the vehicle odometer value
- **a flag indicating whether the position has been authenticated.**

(110a) For places where the daily work period begins or ends entered during the manual entry procedure at card insertion in accordance with requirement 61, the current odometer value and position of the vehicle shall be stored.

(111) The data memory shall be able to hold places and positions where daily work periods begin, end and/or where 3 hours ~~continuous~~ **accumulated** driving time is reached for at least 365 days.

(112) When storage capacity is exhausted, new data shall replace oldest data.

3.12.6 Odometer data

(113) The ~~recording equipment~~ **control device** shall record in its data memory the vehicle odometer value and the corresponding date at midnight every calendar day.

(114) The data memory shall be able to store midnight odometer values for at least 365 calendar days

(115) When storage capacity is exhausted, new data shall replace oldest data.

3.12.7 Detailed speed data

(116) The ~~recording equipment~~ **control device** shall record and store in its data memory the instantaneous speed of the vehicle and the corresponding date and time at every second of at least the last 24 hours that the vehicle has been driven.

3.12.8 Events data

For the purpose of this subparagraph, time shall be recorded with a resolution of 1 second.

(117) The ~~recording equipment~~ **control device** shall record and store in its data memory the following data for each event detected according to the following storage rules:

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Insertion of a non-valid card	- the 10 most recent events.	- date and time of event, - card(s) type, number, issuing Member State Contracting Party and generation of the card creating the event. - number of similar events that day
Card conflict	- the 10 most recent events.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of the two cards creating the conflict.
Driving without an appropriate card	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Card insertion while driving	- the last event for each of the 10 last days of occurrence,	- date and time of the event, - card(s) type, number, issuing Member State Contracting Party and generation, - number of similar events that day
Last card session not correctly closed	- the 10 most recent events.	- date and time of card insertion, - card(s) type, number, issuing Member State Contracting Party and generation, - last session data as read from the card: - date and time of card insertion, VRN, Contracting Party of registration and VU generation.
Over speeding (1)	- the most serious event for each of the 10 last days of occurrence (i.e. the one with the highest average speed), - the 5 most serious events over the last 365 days. - the first event having occurred after the last calibration	- date and time of beginning of event, - date and time of end of event, - maximum speed measured during the event, - arithmetic average speed measured during the event, - card type, number, issuing Member State Contracting Party and generation of the driver card (if applicable), - number of similar events that day.
Power supply interruption (2)	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Communication error with the remote communication facility	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Absence of position information from GNSS receiver	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Communication error with the external GNSS facility	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Motion data error	<ul style="list-style-type: none"> - the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> - date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Vehicle motion conflict	<ul style="list-style-type: none"> - the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> - date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Security breach attempt	<ul style="list-style-type: none"> - the 10 most recent events per type of event. 	<ul style="list-style-type: none"> - date and time of beginning of event, - date and time of end of event (if relevant), - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - type of event.
Time conflict	<ul style="list-style-type: none"> - the longest most serious event for each of the 10 last days of occurrence (i.e. the ones with the greatest difference between control device date and time, and GNSS date and time), - the 5 longest most serious events over the last 365 days. 	<ul style="list-style-type: none"> - recording equipment control device date and time - GNSS date and time, - card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
GNSS anomaly	<ul style="list-style-type: none"> - the longest events for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> - date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.

(1) The ~~recording equipment~~ **control device** shall also record and store in its data memory:

- the date and time of the last OVER SPEEDING CONTROL,
- the date and time of the first over speeding following this OVER SPEEDING CONTROL,
- the number of over speeding events since the last OVER SPEEDING CONTROL.

(2) These data may be recorded at power supply reconnection only, times may be known with an accuracy to the minute.

3.12.9 Faults data

For the purpose of this subparagraph, time shall be recorded with a resolution of 1 second.

(118) The ~~recording equipment~~ **control device** shall attempt to record and store in its data memory the following data for each fault detected according to the following storage rules:

<i>Fault</i>	<i>Storage rules</i>	<i>Data to be recorded per fault</i>
Card fault	- the 10 most recent driver card faults.	- date and time of beginning of fault, - date and time of end of fault, - card(s) type, number, issuing Member State Contracting Party and generation.
Recording equipment Control device faults	- the 10 most recent faults for each type of fault, - the first fault after the last calibration.	- date and time of beginning of fault, - date and time of end of fault, - type of fault, - card(s) type, number and issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the fault.

3.12.10 Calibration data

(119) The ~~recording equipment~~ **control device** shall record and store in its data memory data relevant to:

- known calibration parameters at the moment of activation,
- its very first calibration following its activation,
- its first calibration in the current vehicle (as identified by its VIN),
- the 20 most recent calibrations (if several calibrations happen within one calendar day, only the first and the last one of the day shall be stored).

(120) The following data shall be recorded for each of these calibrations:

- purpose of calibration (activation, first installation, installation, periodic inspection),
- workshop name and address,
- workshop card number, card issuing ~~Member State~~ **Contracting Party** and card expiry date,
- vehicle identification,
- parameters updated or confirmed: w, k, l, tyre size, speed limiting device setting, odometer (old and new values), date and time (old and new values),
- the types and identifiers of all the seals in place
- **the serial numbers of the motion sensor, the external GNSS facility (if any), and the external remote communication facility (if any),**
- **the by-default load type associated to the vehicle (load of either goods or passengers),**
- **the country in which the calibration has been performed, and the date time when the position used to determine this country was provided by the GNSS receiver.**

(121) In addition, the ~~recording equipment~~ **control device** shall record and store in its data memory its ability to use first generation tachograph cards (still activated or not).

(122) The motion sensor shall record and store in its memory the following motion sensor installation data:

- first pairing with a VU (date, time, VU approval number, VU serial number),
- last pairing with a VU (date, time, VU approval number, VU serial number).

(123) The external GNSS facility shall record and store in its memory the following external GNSS facility installation data:

- first coupling with a VU (date, time, VU approval number, VU serial number),
- last coupling with a VU (date, time, VU approval number, VU serial number).

3.12.11 Time adjustment data

(124) The ~~recording equipment~~ **control device** shall record and store in its data memory data relevant to time adjustments performed in calibration mode outside the frame of a regular calibration (def. f):

- the most recent time adjustment,
- the 5 largest time adjustments.

(125) The following data shall be recorded for each of these time adjustments:

- date and time, old value,
- date and time, new value,
- workshop name and address,
- workshop card number, card issuing ~~Member State~~ **Contracting Party**, card generation and card expiry date.

3.12.12 Control activity data

(126) The ~~recording equipment~~ **control device** shall record and store in its data memory the following data relevant to the 20 most recent control activities:

- date and time of the control,
- control card number, card issuing ~~Member State~~ **Contracting Party** and card generation,
- type of the control (displaying and/or printing and/or VU downloading and/or card downloading and/or roadside calibration checking).

(127) In case of downloading, the dates of the oldest and of the most recent days downloaded shall also be recorded.

3.12.13 Company locks data

(128) The ~~recording equipment~~ **control device** shall record and store in its data memory the following data relevant to the 255 most recent company locks:

- lock-in date and time,
- lock-out date and time,

- company card number, card issuing ~~Member State~~ **Contracting Party** and card generation,
- company name and address.

Data previously locked by a lock removed from memory due to the limit above, shall be treated as not locked.

3.12.14 Download activity data

(129) The ~~recording equipment~~ **control device** shall record and store in its data memory the following data relevant to the last data memory downloading to external media while in company or in calibration mode:

- date and time of downloading,
- company or workshop card number, card issuing ~~Member State~~ **Contracting Party** and card generation,
- company or workshop name.

3.12.15 Specific conditions data

(130) The ~~recording equipment~~ **control device** shall record in its data memory the following data relevant to specific conditions:

- date and time of the entry,
- type of specific condition.

(131) The data memory shall be able to hold specific conditions data for at least 365 days (with the assumption that on average, 1 condition is opened and closed per day). When storage capacity is exhausted, new data shall replace oldest data.

3.12.16 Tachograph card data

(132) The ~~recording equipment~~ **control device** shall be able to store the following data related to the different tachograph cards in which had been used in the VU:

- - the tachograph card number and its serial number,
- - the manufacturer of the tachograph card,
- - the tachograph card type,
- - the tachograph card version.

(133) The ~~recording equipment~~ **control device** shall be able to store at least 88 such records.

3.12.17 Border crossings

(133a) The recording equipment shall record and store in its data memory the following information about border crossings:

- **the country that the vehicle is leaving,**
- **the country that the vehicle is entering,**
- **the position where the vehicle has crossed the border.**

(133b) Together with countries and position, the recording equipment shall record and store in its data memory:

- the (co)-driver card number and card issuing Contracting Party,
- the card generation,
- the related GNSS accuracy, date and time,
- a flag indicating whether the position has been authenticated
- the vehicle odometer value at the time of border crossing detection.

(133c) The data memory shall be able to hold border crossings for at least 365 days.

(133d) When storage capacity is exhausted, new data shall replace oldest data.

3.12.18 Load/Unload operations

- (133e) The recording equipment shall record and store in its data memory the following information about load and unload operations of the vehicle: the type of operation (load, unload or simultaneous load/unload),
- the position where the load/unload operation has occurred.

(133f) When the position of the vehicle is not available from the GNSS receiver at the time of the load or unload operation, the recording equipment shall use the latest available position, and the related date and time.

(133g) Together with the type of operation and position, the recording equipment shall record and store in its data memory:

- the driver and/or co-driver card number and card issuing Contracting Party,
- the card generation,
- the date and time of the load/unload operation,
- the related GNSS accuracy, date and time if applicable,
- a flag indicating whether the position has been authenticated,
- the vehicle odometer value.

(133h) The data memory shall be able to store load/unload operations for at least 365 calendar days.

(133i) When storage capacity is exhausted, new data shall replace oldest data.

3.12.19 Digital map

(133j) For the purpose of recording the position of the vehicle when the border of a country is crossed, the recording equipment shall store in its data memory a digital map.

(133k) Allowed digital maps for supporting the border crossing monitoring function of the recording equipment shall be made available by the laboratory in charge of the interoperability tests for download from a dedicated secured website, under various formats

(133l) For each of these maps, a version identifier and a hash value shall be available on the website.

(133m) The maps shall feature:

- a level of definition corresponding to NUTS level 0, according to the Nomenclature of Territorial Units for Statistics,

- a scale of 1:1 million.

(133n) Tachograph manufacturers shall select a map from the website and download it securely.

(133o) Tachograph manufacturers shall only use a downloaded map from the website after having verified its integrity using the hash value of the map.

(133p) The selected map shall be imported in the recording equipment by its manufacturer, under an appropriate format, but the semantic of the imported map shall remain unchanged.

(133q) The manufacturer shall also store the version identifier of the map used in the recording equipment.

(133r) It shall be possible to update or replace the stored digital map by a new one made available by the laboratory competent for interoperability tests.

(133s) Digital map updates shall be made using the software update mechanisms set up by the manufacturer, in application of requirements 226d and 226e, so that the recording equipment can verify the authenticity and integrity of a new imported map, before storing it and replacing the previous one.

(133t) Tachograph manufacturers may add additional information to the basic map referred to in requirement (133m), for purposes other than recording border crossings, such as the borders of the regions of the Contracting Parties, provided that the semantic of the basic map is not changed.

3.13 Reading from tachograph cards

(134) The ~~recording equipment~~ **control device** shall be able to read from first and second generation tachograph cards, where applicable, the necessary data:

- to identify the card type, the card holder, the previously used vehicle, the date and time of the last card withdrawal and the activity selected at that time,
- to check that last card session was correctly closed,
- to compute the driver's continuous driving time, cumulative break time and accumulated driving times for the previous and the current week,
- to print requested printouts related to data recorded on a driver card,
- to download a driver card to external media.

This requirement only applies to first generation tachograph cards if their use has not been suppressed by a workshop.

(135) In case of a reading error, the ~~recording equipment~~ **control device** shall try again, three times maximum, the same read command, and then if still unsuccessful, declare the card faulty and non-valid.

(135a) The structure in the "TACHO_G2" application depends on the version. Version 2 cards contain additional Elementary Files to the ones of version 1 cards, in particular:

- in driver and workshop cards:
 - EF Places_Authentication shall contain the authentication status of the vehicle positions stored in EF Places. A timestamp shall be stored with each authentication status, which shall be exactly the same as the date and time of the entry stored with the corresponding position in EF Places.

- **EF GNSS_Places_Authentication** shall contain the authentication status of the vehicle positions stored in **EF GNSS_Places**. A timestamp shall be stored with each authentication status, which shall be exactly the same as the date and time of the entry stored with the corresponding position in **EF Places**.
- **EF Border_Crossings**, **EF Load_Unload_Operations** and **EF Load_Type_Entries** shall contain data related to border crossings, load/unload operations and load types.
- in workshop cards:
 - **EF Calibration_Add_Data** shall contain additional calibration data to the ones stored in **EF Calibration**. The old date and time value and the vehicle identification number shall be stored with each additional calibration data record, which shall be exactly the same as the old date and time value and the vehicle identification number stored with the corresponding calibration data in **EF Calibration**.
- in all tachograph cards:
 - **EF VU_Configuration** shall contain the card owner tachograph specific settings.

The vehicle unit shall ignore any authentication status found in **EF Places_Authentication** or **EF GNSS_Places_Authentication**, when no vehicle position with the same timestamp is found in **EF Places** or **EF GNSS_Places**.

The vehicle unit shall ignore the elementary file **EF VU_Configuration** in all cards insofar as no specific rules have been provided with respect to the use of such elementary file. Those rules shall be set out through an amendment of Appendix 1C, which shall include the modification or deletion of this paragraph.

3.14. Recording and storing on tachograph cards

3.14.1 Recording and storing in first generation tachograph cards

(136) Provided first generation tachograph cards use has not been suppressed by a workshop, the ~~recording equipment~~ **control device** shall record and store data exactly in the same way as a first generation ~~recording equipment~~ **control device** would do.

(137) The ~~recording equipment~~ **control device** shall set the “card session data” in the driver or workshop card right after the card insertion.

(138) The ~~recording equipment~~ **control device** shall update data stored on valid driver, workshop, company and/or control cards with all necessary data relevant to the period while the card is inserted and relevant to the card holder. Data stored on these cards are specified in Chapter 4.

(139) The ~~recording equipment~~ **control device** shall update driver activity and places data (as specified in 4.5.3.1.9 and 4.5.3.1.11), stored on valid driver and/or workshop cards, with activity and places data manually entered by the cardholder.

(140) All events **and faults** not defined for the first generation ~~recording equipment~~ **control device**, shall not be stored on the **first generation** driver and workshop cards.

(141) Tachograph cards data update shall be such that, when needed and taking into account card actual storage capacity, most recent data replace oldest data.

(142) In the case of a writing error, the ~~recording equipment~~ **control device** shall try again, three times maximum, the same write command and then if still unsuccessful, declare the card faulty and non-valid.

(143) Before releasing a driver **or workshop** card and after all relevant data have been stored on the card, the ~~recording equipment~~ **control device** shall reset the “card session data”.

3.14.2 Recording and storing in second generation tachograph cards

(144) Second generation tachograph cards shall contain 2 different card applications, the first of which shall be exactly the same as the TACHO application of first generation tachograph cards, and the second the “TACHO_G2” application, as specified in Chapter 4 and ~~Appendix~~ **Sub-appendix 2**.

The structure in the ‘TACHO_G2’ application depends on the version. Version 2 cards contain additional Elementary Files to the ones of version 1 cards.

(145) The ~~recording equipment~~ **control device** shall set the “card session data” in the driver or workshop card right after the card insertion.

(146) The ~~recording equipment~~ **control device** shall update data stored on the 2 card applications of valid driver, workshop, company and/or control cards with all necessary data relevant to the period while the card is inserted and relevant to the card holder. Data stored on these cards are specified in Chapter 4.

(147) The ~~recording equipment~~ **control device** shall update driver activity places and positions data (as specified in 4.5.3.1.9, 4.5.3.1.11, 4.5.3.2.9 and 4.5.3.2.11), stored on valid driver and/or workshop cards, with activity and places data manually entered by the cardholder.

(147a) On insertion of a driver or workshop card, the recording equipment shall store on the card the by-default load type of the vehicle.

(147b) On insertion of a driver or workshop card, and after the manual entry procedure, the recording equipment shall check the last place where the daily work period begins or ends stored on the card. This place may be temporary, as specified in requirement 59. If this place is in a different country from the current one in which the vehicle is located, the recording equipment shall store on the card a border crossing record, with:

- **the country that the driver left: not available,**
- **the country that the driver is entering: the current country in which the vehicle is located,**
- **the date and time when the driver has crossed the border: the card insertion time,**
- **the position of the driver when the border has been crossed: not available,**
- **the vehicle odometer value: not available.**

(148) Tachograph cards data update shall be such that, when needed and taking into account card actual storage capacity, most recent data replace oldest data.

(149) In the case of a writing error, the ~~recording equipment~~ **control device** shall try again, three times maximum, the same write command and then if still unsuccessful, declare the card faulty and non-valid.

(150) Before releasing a driver card and after all relevant data have been stored on the 2 card applications of the card, the ~~recording equipment~~ **control device** shall reset the “card session data”.

(150a) The vehicle unit shall ignore the elementary file EF VU_Configuration in all cards insofar as no specific rules have been provided with respect to the use of such elementary file. Those rules shall be set out through an amendment of Appendix 1C, which shall include the modification or deletion of this paragraph.

3.15. Displaying

(151) The display shall include at least 20 characters.

(152) The minimum character size shall be 5 mm high and 3.5 mm wide.

(153) The display shall support the characters specified in ~~Appendix~~ **Sub-appendix 1** Chapter 4 'Character sets'. The display may use simplified glyphs (e.g. accented characters may be displayed without accent, or lower case letters may be shown as upper case letters).

(154) The display shall be provided with adequate non-dazzling lighting.

(155) Indications shall be visible from outside the control device.

(156) The ~~recording equipment~~ **control device** shall be able to display:

- default data,
- data related to warnings,
- data related to menu access,
- other data requested by a user.
- Additional information may be displayed by the ~~recording equipment~~ **control device**, provided that it is clearly distinguishable from information required above.

(157) The display of the ~~recording equipment~~ **control device** shall use the pictograms or pictograms combinations listed in ~~Appendix~~ **Sub-appendix 3**. Additional pictograms or pictograms combinations may also be provided by the display, if clearly distinguishable from the aforementioned pictograms or pictograms combinations.

(158) The display shall always be ON when the vehicle is moving.

(159) The ~~recording equipment~~ **control device** may include a manual or automatic feature to turn the display OFF when the vehicle is not moving.

Displaying format is specified in ~~Appendix~~ **Sub-appendix 5**.

3.15.1 Default display

(160) When no other information needs to be displayed, the ~~recording equipment~~ **control device** shall display, by default, the following:

- the local time (as a result of UTC time + offset as set by the driver),
- the mode of operation,
- the current activity of the driver and the current activity of the co-driver,
- information related to the driver:
 - if his current activity is DRIVING, his current continuous driving time and his current cumulative break time,
 - if his current activity is not DRIVING, the current duration of this activity (since it was selected) and his current cumulative break time.

(161) Display of data related to each driver shall be clear, plain and unambiguous. In the case where the information related to the driver and the co-driver cannot be displayed at the same time, the ~~recording equipment~~ **control device** shall display by default the information related to the driver and shall allow the user to display the information related to the co-driver.

(162) In the case where the display width does not allow displaying by default the mode of operation, the ~~recording equipment~~ **control device** shall briefly display the new mode of operation when it changes.

(163) The ~~recording equipment~~ **control device** shall briefly display the card holder name at card insertion.

(164) When an “OUT OF SCOPE” or FERRY/TRAIN condition is opened, then the default display must show using the relevant pictogram that the particular condition is opened (it is acceptable that the driver’s current activity may not be shown at the same time).

3.15.2 Warning display

(165) The ~~recording equipment~~ **control device** shall display warning information using primarily the pictograms of ~~Appendix~~ **Sub-appendix** 3, completed where needed by additional numerically coded information. A literal description of the warning may also be added in the driver’s preferred language.

3.15.3 Menu access

(166) The ~~recording equipment~~ **control device** shall provide necessary commands through an appropriate menu structure.

3.15.4 Other displays

(167) It shall be possible to display selectively on request:

- the UTC date and time, and local time offset,
- the content of any of the ~~six~~ printouts **listed in requirement 169** under the same formats as the printouts themselves,
- the continuous driving time and ~~cumulative~~ **accumulative** break time of the driver,
- the continuous driving time and ~~cumulative~~ **accumulative** break time of the co-driver,
- the ~~emulated~~ **accumulated** driving time of the driver for the previous and the current week,
- the ~~emulated~~ **accumulated** driving time of the co-driver for the previous and the current week,
- optional:
 - the current duration of co-driver activity (since it was selected),
 - the ~~emulated~~ **accumulated** driving time of the driver for **the** current week,
 - the ~~emulated~~ **accumulated** driving time of the co-driver for the current daily work period,
 - the accumulated driving time of the driver for the current daily work period.

(168) Printout content display shall be sequential, line by line. If the display width is less than 24 characters the user shall be provided with the complete information through an

appropriate mean (several lines, scrolling, ...). **Printout lines devoted to hand-written information may be omitted for display.** ~~Printout lines devoted to hand-written information may be omitted for display~~

3.16. Printing

(169) The ~~recording equipment~~ **control device** shall be able to print information from its data memory and/or from tachograph cards in accordance with the seven following printouts:

- driver activities from card daily printout,
- driver activities from Vehicle Unit daily printout,
- events and faults from card printout,
- events and faults from Vehicle Unit printout,
- technical data printout,
- over speeding printout.
- tachograph card data history for a given VU (see chapter 3.12.16)
- The detailed format and content of these printouts are specified in ~~Appendix~~ **Sub-appendix 4.**

Additional data may be provided at the end of the printouts.

Additional printouts may also be provided by the ~~recording equipment~~ **control device**, if clearly distinguishable from the seven aforementioned printouts.

(170) The “driver activities from card daily printout” and “Events and faults from card printout” shall be available only when a driver card or a workshop card is inserted in the ~~recording equipment~~ **control device**. ~~The recording equipment~~ **The control device** shall update data stored on the relevant card before starting printing.

(171) In order to produce the “driver activities from card daily printout” or the “events and faults from card printout”, the ~~recording equipment~~ **control device** shall:

- either automatically select the driver card or the workshop card if one only of these cards is inserted,
- or provide a command to select the source card or select the card in the driver slot if two of these cards are inserted in the ~~recording equipment~~ **control device**.

(172) The printer shall be able to print 24 characters per line.

(173) The minimum character size shall be 2.1 mm high and 1.5 mm wide.

(174) The printer shall support the characters specified in ~~Appendix~~ **Sub-appendix 1** Chapter 4 ‘Character sets’.

(175) Printers shall be so designed as to produce these printouts with a degree of definition likely to avoid any ambiguity when they are read.

(176) Printouts shall retain their dimensions and recordings under normal conditions of humidity (10-90%) and temperature.

(177) The type approved paper used by the ~~recording equipment~~ **control device** shall bear the relevant type approval mark and an indication of the type(s) of ~~recording equipment~~ **control device** with which it may be used.

(178) Printouts shall remain clearly legible and identifiable under normal conditions of storage, in terms of light intensity, humidity and temperature, for at least two years.

(179) Printouts shall conform at least to the test specifications defined in ~~Appendix~~ **Sub-appendix 9**.

(180) It shall also be possible to add hand-written notes, such as the driver's signature, to these documents.

(181) The ~~recording equipment~~ **control device** shall manage "paper out" events while printing by, once paper has been re-loaded, restarting printing from printout beginning or by continuing printing and providing an unambiguous reference to previously printed part.

3.17. Warnings

(182) The ~~recording equipment~~ **control device** shall warn the driver when detecting any event and/or fault.

(183) Warning of a power supply interruption event may be delayed until the power supply is reconnected.

(184) The ~~recording equipment~~ **control device** shall warn the driver 15 minutes before and at the time of exceeding the maximum allowed continuous driving time.

(185) Warnings shall be visual. Audible warnings may also be provided in addition to visual warnings.

(186) Visual warnings shall be clearly recognisable by the user, shall be situated in the driver's field of vision and shall be clearly legible both by day and by night.

(187) Visual warnings may be built into the ~~recording equipment~~ **control device** and/or remote from the ~~recording equipment~~ **control device**.

(188) In the latter case it shall bear a "T" symbol.

(189) Warnings shall have a duration of at least 30 seconds, unless acknowledged by the user by hitting one or more specific keys of the ~~recording equipment~~ **control device**. This first acknowledgement shall not erase warning cause display referred to in next paragraph.

(190) Warning cause shall be displayed on the ~~recording equipment~~ **control device** and remain visible until acknowledged by the user using a specific key or command of the ~~recording equipment~~ **control device**.

(191) Additional warnings may be provided, as long as they do not confuse drivers in relation to previously defined ones.

3.18. Data downloading to external media

(192) The ~~recording equipment~~ **control device** shall be able to download on request data from its data memory or from a driver card to external storage media via the calibration/downloading connector. The ~~recording equipment~~ **control device** shall update data stored on the relevant card before starting downloading.

(193) In addition and as an optional feature, the ~~recording equipment~~ **control device** may, in any mode of operation, download data through any ~~another~~ **interfacemeans** to a company authenticated through this channel. In such a case, company mode data access rights shall apply to this download.

(194) Downloading shall not alter or delete any stored data.

195) The calibration/downloading connector electrical interface is specified in ~~Appendix~~ **Sub-appendix 6**.

(196) Downloading protocols are specified in ~~Appendix~~ **Sub-appendix 7**.

(196a) A transport undertaking which uses vehicles that are fitted with recording equipment complying with this Appendix shall ensure that all data are downloaded from the vehicle unit and driver cards.

The maximum period within which the relevant data are downloaded shall not exceed:

- **90 days for data from vehicle unit;**
- **28 days for data from the driver card.**

(196b) Transport undertakings shall keep the data downloaded from the vehicle unit and driver cards for at least twelve months following recording.

3.19. Remote communication for targeted roadside checks

(197) When the ignition is on, the Vehicle Unit shall store every 60 seconds in the remote communication facility the most recent data necessary for the purpose of targeted roadside checks. Such data shall be encrypted and signed as specified in ~~Appendix~~ **Sub-appendix 11** and ~~Appendix~~ **Sub-appendix 14**.

(198) Data to be checked remotely shall be available to remote communication readers through wireless communication, as specified in ~~Appendix~~ **Sub-appendix 14**.

(199) Data necessary for the purpose of targeted roadside checks shall be related to :

- the latest security breach attempt,
- the longest power supply interruption,
- sensor fault,
- motion data error,
- vehicle motion conflict,
- driving without a valid card,
- card insertion while driving,
- time adjustment data,
- calibration data including the dates of the two latest stored calibration records,
- vehicle registration number,
- speed recorded by the tachograph,
- **vehicle position**
- **an indication if the driver may currently infringe the driving times.**

3.20. ~~Output of~~ Data exchanges with ~~to~~ additional external devices

(200) The ~~recording equipment control device may~~ shall also be equipped with **an ITS standardised interfaces in accordance with Sub-appendix 13**, allowing the data recorded or produced by **either the tachograph or the tachograph cards** to be used ~~in operational or calibration mode,~~ by an external facility.

In operational mode, the driver consent shall be needed for the transmission of personal data through the ITS interface. Nevertheless, the driver consent shall not apply to tachograph or card data accessed in control, company or calibration mode. The data and functional access rights for these modes are specified in requirements 12 and 13.

The following requirements shall apply to ITS data made available through that interface:

- personal data shall only be available after the verifiable consent of the driver has been given, accepting that personal data can leave the vehicle network.

A set of selected existing data that can be available via the ITS interface, and the classification of the data as personal or not personal are specified in Sub-appendix 13. Additional data may also be output in addition to the set of data provided in Sub-appendix 13. The VU manufacturer shall classify those data as “personal” or “not personal”, being the driver consent applicable to those data classified as “personal”,

- at any moment, the driver consent can be enabled or disabled through commands in the menu, provided the driver card is inserted,
- in any circumstances, the presence of the ITS interface shall not disturb or affect the correct functioning and the security of the vehicle unit.

Additional vehicle unit interfaces may co-exist, provided they fully comply with the requirements of Sub-appendix 13 in terms of driver consent. The recording equipment shall have the capacity to communicate the driver consent status to other platforms in the vehicle network and to external devices.

For personal data injected in the vehicle network, which are further processed outside the vehicle network, it shall not be the responsibility of the tachograph manufacturer to have that personal data process compliant with the applicable legislation in the EU regarding data protection.

The ITS interface shall also allow for data entry during the manual entry procedure in accordance with requirement 61, for both the driver and the co-driver.

The ITS interface may also be used to enter additional information, in real time, such as:

- driver activity selection, in accordance with requirement 46,
- places in accordance with requirement 56,
- specific conditions, in accordance with requirement 62,
- load/unload operations, in accordance with requirement 62a.

This information may also be entered through other interfaces.

~~In Appendix **Sub-appendix** 13, an optional ITS interface is specified and standardized. Other similar interfaces may co exist, provided they fully comply with the requirements of appendix **Sub-appendix** 13 in term of minimum list of data, security and driver consent.~~

~~The driver consent does not apply to data transmitted by the control device to the vehicle network. In case the personal data injected in the vehicle network are further processed outside the vehicle network, it is the responsibility of the vehicle manufacturer to have that personal data process compliant with the legislation on personal data protection applicable in the territory of the Contracting Parties and with the Convention for the protection of individuals with regard to automatic processing of personal data. The driver consent doesn't apply either to tachograph data downloaded~~

~~to a remote company (requirement 193), as this scenario is monitored by the company card access right.~~

The following requirements apply to ITS data made available through that interface:

- ~~— these data are a set of selected existing data from the tachograph data dictionary (Appendix **Sub-appendix 1**),~~
- ~~— a subset of these selected data are marked ‘personal data’,~~
- ~~— the subset of ‘personal data’ is only available if the verifiable consent of the driver, accepting his personal data can leave the vehicle network, is enabled,~~
- ~~— At any moment, the driver consent can be enabled or disabled through commands in the menu, provided the driver card is inserted,~~
- ~~— the set and subset of data will be broadcasted via Bluetooth wireless protocol in the radius of the vehicle cab, with a refresh rate of 1 minute,~~
- ~~— the pairing of the external device with the ITS interface will be protected by a dedicated and random PIN of at least 4 digits, recorded in and available through the display of each vehicle unit,~~
- ~~— in any circumstances, the presence of the ITS interface cannot disturb or affect the correct functioning and the security of the vehicle unit.~~

~~Other data may also be output in addition to the set of selected existing data, considered as the minimum list, provided they cannot be considered as personal data.~~

~~The recording equipment **control device** notify other external facilities about shall **have** the consent of **capacity to communicate the driver consent status to other platforms in the vehicle network.**~~

~~When the ignition of the vehicle is ON, these data shall be permanently broadcasted.~~

(201) The serial link interface as specified in Annex **Appendix 1B to Regulation (EEC) N° 3821/85 this Agreement**, as last amended, can continue to equip tachographs for back compatibility. **The serial link is classified as a part of the vehicle network, in accordance with requirement 200.**~~Anyhow, the driver consent is still required in case personal data are transmitted.~~

3.21. Calibration

(202) The calibration function shall allow:

- to automatically pair the motion sensor with the VU,
- to automatically couple the external GNSS facility with the VU if applicable,
- to digitally adapt the constant of the ~~recording equipment~~ **control device** (k) to the characteristic coefficient of the vehicle (w) ,
- to adjust the current time within the validity period of the inserted workshop card,
- to adjust the current odometer value,
- to update motion sensor identification data stored in the data memory,
- to update, if applicable, external GNSS facility identification data stored in the data memory,
- to update the types and identifiers of all the seals in place,

- to update or confirm other parameters known to the ~~recording equipment~~ **control device**: vehicle identification, w, l, tyre size and speed limiting device setting if applicable, **and by-default load type**,-
- **to automatically store the country in which the calibration has been performed, and the date time when the position used to determine this country was provided by the GNSS receiver.**

(203) In addition, the calibration function shall allow to suppress the use of first generation tachograph cards in the ~~recording equipment~~ **control device**, provided the conditions specified in ~~Appendix~~ **Sub-appendix 15** are met.

(204) Pairing the motion sensor to the VU shall consist, at least, in:

updating motion sensor installation data held by the motion sensor (as needed),

copying from the motion sensor to the VU data memory the necessary motion sensor identification data.

(205) Coupling the external GNSS facility to the VU shall consist, at least, in:

- updating external GNSS facility installation data held by the external GNSS facility (as needed),
- copying from the external GNSS facility to the VU data memory the necessary external GNSS facility identification data including the serial number of the external GNSS facility,-

~~The coupling shall be followed by the verification of the GNSS position information.~~

(206) The calibration function shall be able to input necessary data through the calibration/downloading connector in accordance with the calibration protocol defined in ~~Appendix~~ **Sub-appendix 8**. The calibration function may also input necessary data through other means.

3.22. Roadside calibration checking

(207) The roadside calibration checking function shall allow reading the motion sensor serial number (possibly embedded in the adaptor) and the external GNSS facility serial number (when applicable), connected to the vehicle unit, at the time of the request.

(208) This reading shall at least be possible on the vehicle unit display through commands in the menus.

(209) The roadside calibration checking function shall also allow controlling the selection of the I/O mode of the calibration I/O signal line specified in ~~Appendix~~ **Sub-appendix 6**, via the K-line interface. This shall be done through the ECUAdjustmentSession, as specified in ~~Appendix~~ **Sub-appendix 8**, section 7 Control of Test Pulses – Input output control functional unit.

When the I/O mode of the calibration I/O signal line is active according to this requirement, the “Driving without an appropriate card” warning (requirement 75) shall not be triggered by the vehicle unit.

3.23. Time adjustment

(210) The time adjustment function shall allow for automatically adjusting the current time. Two time sources are used in the ~~recording equipment~~ **control device** for time adjustment: 1) the internal VU clock, 2) the GNSS receiver.

(211) The time setting of the VU internal clock shall be automatically re-adjusted ~~at intervals of~~ **at variable time intervals every 12 hours maximum. The next automatic time re-adjustment shall be triggered between 72h and 168h after the previous one, and after the VU can access to GNSS time through a valid authenticated position message in accordance with Sub-appendix 12. Nevertheless, the time adjustment shall never be bigger than the accumulated maximal time drift per day, as calculated by the VU manufacturer in accordance with requirement 41b. If the difference between internal VU clock time and GNSS receiver time is bigger than the accumulated maximum time drift per day, then the time adjustment shall bring the VU internal clock as close as possible to the GNSS receiver time. The time setting may only be done if the time provided by the GNSS receiver is obtained using authenticated position messages as set out in Sub-appendix 12** ~~When this delay has expired and re-adjustment is not possible because the GNSS signal is not available, the time setting shall be done as soon as the VU can access a valid time provided by GNSS receiver, according to the vehicle ignition conditions. The time reference for the automatic time setting of the VU internal clock shall be the time provided in the authenticated position message derived from the GNSS receiver. A time conflict event shall be triggered if the current time deviates more than one (1) minute from the time information provided by the GNSS receiver.~~

(212) The time adjustment function shall also allow for triggered adjustment of the current time, in calibration mode.

Workshops may adjust time:

- **either by writing a time value in the VU, using the WriteDataByIdentifier service in accordance with section 6.2 of Sub-appendix 8,**
- **or by requesting an alignment of the VU clock to the time provided by the GNSS receiver. This may only be done if the time provided by the GNSS receiver is obtained using authenticated position messages. In this latter case, the RoutineControl service shall be used in accordance with section 8 of Sub-appendix 8.**

3.24. Performance characteristics

(213) The Vehicle Unit shall be fully operational in the temperature range -20°C to 70°C, the external GNSS facility in the temperature range -20°C to 70°C, and the motion sensor in the temperature range -40°C to 135°C. Data memory content shall be preserved at temperatures down to -40°C.

(214) The tachograph shall be fully operational in the humidity range 10% to 90%.

(215) The seals used in the smart tachograph shall withstand the same conditions than those applicable to the tachograph components to which they are affixed.

(216) The ~~recording equipment~~ **control device** shall be protected against over-voltage, inversion of its power supply polarity, and short circuits.

(217) Motion sensors shall either:

- react to a magnetic field disturbing vehicle motion detection. In such circumstances, the vehicle unit will record and store a sensor fault (requirement 88) or,
- have a sensing element that is protected from, or immune to, magnetic fields.

(218) The ~~recording equipment~~ **control device** and the external GNSS facility shall conform to international ~~regulation~~ **UNECE R10 Regulation 10** and shall be protected against electrostatic discharges and transients.

3.25. Materials

(219) All the constituent parts of the ~~recording equipment~~ **control device** shall be made of materials of sufficient stability and mechanical strength and with stable electrical and magnetic characteristics.

(220) For normal conditions of use, all the internal parts of the equipment shall be protected against damp and dust.

(221) The Vehicle Unit and the external GNSS facility shall meet the protection grade IP 40 and the motion sensor shall meet the protection grade IP 64, as per standard IEC 60529:1989 including A1:1999 and A2:2013.

(222) The ~~recording equipment~~ **control device** shall conform to applicable technical specifications related to ergonomic design.

(223) The ~~recording equipment~~ **control device** shall be protected against accidental damage.

3.26. Markings

(224) If the ~~recording equipment~~ **control device** displays the vehicle odometer value and speed, the following details shall appear on its display:

- near the figure indicating the distance, the unit of measurement of distance, indicated by the abbreviation “km”,
- near the figure showing the speed, the entry “km/h”.

The ~~recording equipment~~ **control device** may also be switched to display the speed in miles per hour, in which case the unit of measurement of speed shall be shown by the abbreviation “mph”. The ~~recording equipment~~ **control device** may also be switched to display the distance in miles, in which case the unit of measurement of distance shall be shown by the abbreviation “mi”.

(225) A descriptive plaque shall be affixed to each separate component of the ~~recording equipment~~ **control device** and shall show the following details:

- name and address of the ~~equipment~~ manufacturer,
- manufacturer’s part number and year of manufacture ~~of the equipment~~,
- ~~equipment~~ serial number,
- **type**-approval mark ~~for the equipment type~~.

(226) When physical space is not sufficient to show all above mentioned details, the descriptive plaque shall show at least: the manufacturer’s name or logo, and the ~~equipment’s~~ part number.

3.27. Monitoring border crossings

(226a) This function shall detect when the vehicle has crossed the border of a country, which country has been left and which country has been entered.

(226b) The border crossing detection shall be based on the position measured by the recording equipment, and stored digital map in accordance with point 3.12.19.

(226c) Border crossings related to the presence of the vehicle in a country for a shorter period than 120s shall not be recorded.

3.28 Software update

(226d) The vehicle unit shall incorporate a function for the implementation of software updates whenever such updates do not involve the availability of additional hardware resources beyond the resources set out in requirement 226f, and the type-approval authorities give their authorisation to the software updates based on the existing type-approved vehicle unit in accordance with this Agreement.

(226e) The software update function shall be designed for supporting the following functional features, whenever they are legally required:

- modification of the functions referred to in point 2.2, except the software update function itself,
- the addition of new functions directly related to the enforcement of EU legislation on road transport,
- modification of the modes of operation in point 2.3,
- modification of the file structure such as the addition of new data or the increase of the file size,
- deployment of software patches to address software as well as security defects or reported attacks on the functions of the recording equipment.

(226f) The vehicle unit shall provide free hardware resources of at least 35% for software and data needed for the implementation of requirement 226e and free hardware resources of at least 65% for the update of the digital map based on the hardware resources required for the NUTS 0 map version 2021.

4. Construction and functional requirements for tachograph cards

4.1. Visible data

The front page shall contain:

(227) the words “Driver card” or “Control card” or “Workshop card” or “Company card” printed in capital letters in the official language or languages of the ~~Member State~~ **Contracting Party** issuing the card, according to the type of the card.

(228) the name of the ~~Member State~~ **Contracting Party** issuing the card (optional);

(229) For EU **Member States**, the distinguishing sign of the Member States issuing the card, printed in negative in a blue rectangle and encircled by 12 yellow stars. The distinguishing signs shall be as follows:

B	Belgium	LV	Latvia
BG	Bulgaria	L	Luxembourg
CZ	Czech Republic	LT	Lithuania
CY	Cyprus	M	Malta
DK	Denmark	NL	The Netherlands
D	Germany	A	Austria
EST	Estonia	PL	Poland
GR	Greece	P	Portugal
		RO	Romania
		SK	Slovakia
		SLO	Slovenia
E	Spain	FIN	Finland

B	Belgium	LV	Latvia
BG	Bulgaria	L	Luxembourg
CZ	Czech Republic	LT	Lithuania
CY	Cyprus	M	Malta
F	France	S	Sweden
HR	Croatia		
H	Hungary		
IRL	Ireland	UK	The United Kingdom
I	Italy		

For non-EU Contracting Parties, the distinguishing sign of the Contracting Party issuing the card. The distinguishing signs of non-European Union Contracting Parties are those drawn in accordance with the 1968 Vienna Convention on Road Traffic or the 1949 Geneva Convention on Road Traffic.

(230) information specific to the card issued, numbered as follows:

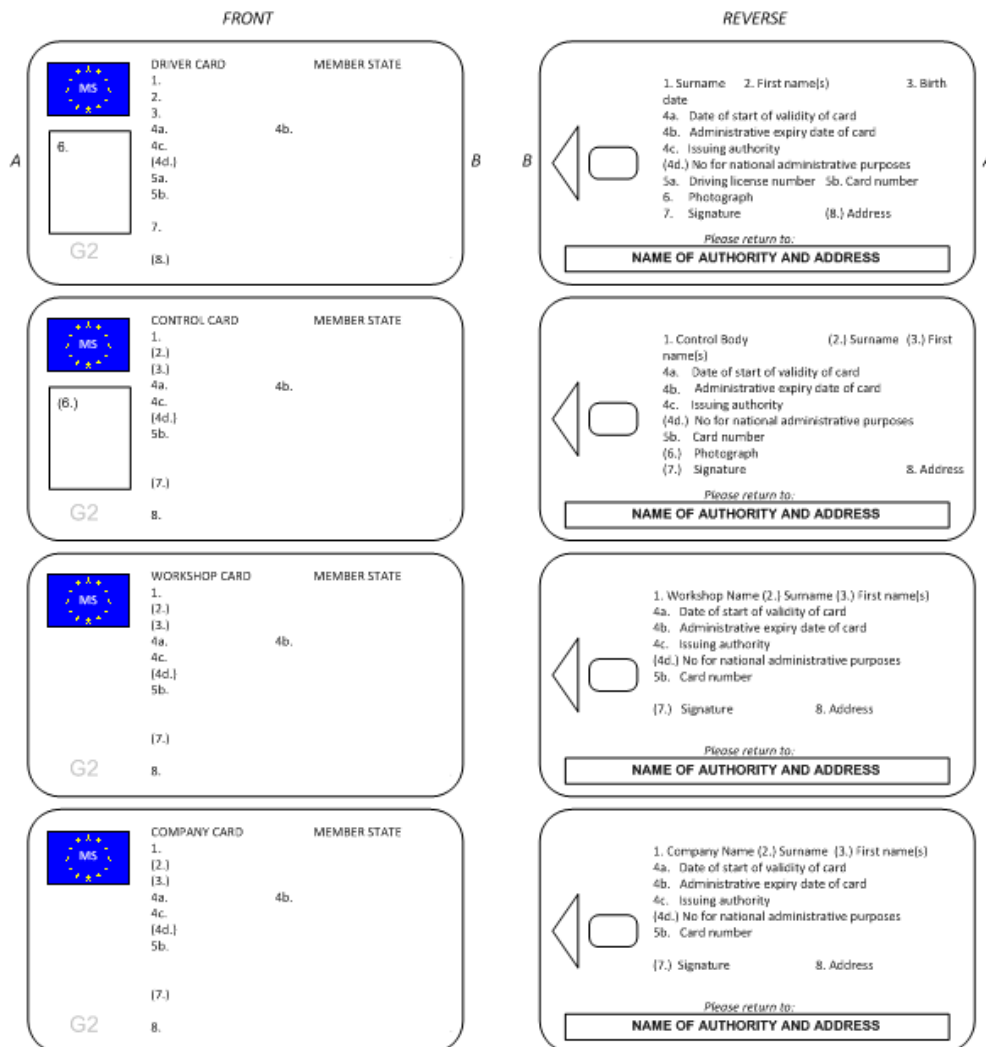
	<i>Driver card</i>	<i>Control Card</i>	<i>Company or Workshop card</i>
1.	surname of the driver	control body name	company or workshop name
2.	first name(s) of the driver	surname of the controller (if applicable)	surname of card holder (if applicable)
3.	birth date of the driver	first name(s) of the controller (if applicable)	first name(s) of card holder (if applicable)
4.a	card start of validity date		
4.b	card expiry date		
4.c	the name of the issuing authority (may be printed on reverse page)		
4.d	a different number from the one under heading 5, for administrative purposes (optional)		
5. a	Driving licence number (at the date of issue of the driver card)	-	-
5. b	Card number		
6.	Photograph of the driver	photograph of the controller (optional)	photograph of the fitter (optional)-
7.	Signature of the holder (optional)		
8.	Normal place of residence, or postal address of the holder (optional).	Postal address of control body	postal address of company or workshop

(231) dates shall be written using a “dd/mm/yyyy” or “dd.mm.yyyy” format (day, month, year).

The reverse page shall contain:

- (232) an explanation of the numbered items which appear on the front page of the card;
- (233) with the specific written agreement of the holder, information which is not related to the administration of the card may also be added, such addition will not alter in any way the use of the model as a tachograph card.
- (234) Tachograph cards shall be printed with the following background predominant colours:
- driver card: white,
 - control card: blue,
 - workshop card: red,
 - company card: yellow.
- (235) Tachograph cards shall bear at least the following features for protection of the card body against counterfeiting and tampering:
- a security design background with fine guilloche patterns and rainbow printing,
 - in the area of the photograph, the security design background and the photograph shall overlap,
 - at least one two-coloured microprint line.

COMMUNITY MODEL TACHOGRAPH CARDS



After consulting the Commission, Member States

Bernardo to verify the need for duplicate templates as per the original

(236) **Contracting Parties** may add colours or markings, such as national symbols and security features, without prejudice to the other provisions of this **Annex Appendix**.

Temporary cards referred to in Article 26.4 of Regulation (EU) N°. 165/2014 shall comply with the provisions of this Annex.

(237) **Reserved.**

4.2. Security

The system security aims at protecting integrity and authenticity of data exchanged between the cards and the ~~recording equipment~~ **control device**, protecting the integrity and authenticity of data downloaded from the cards, allowing certain write operations onto the cards to ~~recording equipment~~ **control device** only, decrypting certain data, ruling out any

possibility of falsification of data stored in the cards, preventing tampering and detecting any attempt of that kind.

(238) In order to achieve the system security, the tachograph cards shall meet the security requirements defined in ~~Appendixes~~ **Sub-appendixes** 10 and 11.

(239) Tachograph cards shall be readable by other equipment such as personal computers.

4.3. Standards

(240) Tachograph cards shall comply with the following standards:

- ISO/IEC 7810 Identification cards – Physical characteristics,
- ISO/IEC 7816 Identification cards - Integrated circuit cards:
 - Part 1: Physical characteristics,
 - Part 2: Dimensions and position of the contacts (ISO/IEC 7816-2:2007),
 - Part 3: Electrical interface and transmission protocols (ISO/IEC 7816-3:2006),
 - Part 4: Organisation, security and commands for interchange (ISO/IEC 7816-4:2013 + Cor 1:2014),
 - Part 6: Interindustry data elements for interchange (ISO/IEC 7816-6:2004 + Cor 1:2006),
 - Part 8: Commands for security operations (ISO/IEC 7816-8:2004).

Tachograph cards shall be tested in accordance to ISO/IEC 10373-3 :2010 Identification cards – Test methods - Part 3: Integrated circuit cards with contacts and related interface devices.

4.4. Environmental and electrical specifications

(241) Tachograph cards shall be capable of operating correctly in all the climatic conditions normally encountered in ~~community~~ **the territory of the Contracting Parties** and at least in the temperature range -25°C to +70°C with occasional peaks of up to +85°C, “occasional” meaning not more than 4 hours each time and not over 100 times during the life time of the card.

(242) Tachograph cards shall be capable of operating correctly in the humidity range 10% to 90%.

(243) Tachograph cards shall be capable of operating correctly for a five-year period if used within the environmental and electrical specifications.

(244) During operation, tachograph cards shall conform to ECE R10, related to electromagnetic compatibility, and shall be protected against electrostatic discharges.

4.5. Data storage

For the purpose of this paragraph,

- times are recorded with a resolution of one minute, unless otherwise specified,
- odometer values are recorded with a resolution of one kilometre,
- speeds are recorded with a resolution of 1 km/h,

- positions (latitudes and longitudes) are recorded in degrees and minutes with a resolution of 1/10 of minute.

The tachograph cards functions, commands and logical structures, fulfilling data storage requirements are specified in ~~Appendix~~ **Sub-appendix 2**.

If not otherwise specified, data storage on tachograph cards shall be organized in such a way, that new data replaces stored oldest data in case the foreseen memory size for the particular records is exhausted.

(245) This paragraph specifies minimum storage capacity for the various application data files. Tachograph cards shall be able to indicate to the ~~recording equipment control device~~ the actual storage capacity of these data files.

(246) Any additional data that may be stored on tachograph cards, **provided that the storage of those data complies with the applicable legislation regarding data protection**~~related to other applications possibly borne by the card, shall be stored in accordance with Directive 95/46/EC the legislation on personal data protection applicable in the territory of 24 October 1995 on the Contracting Parties and with the Convention for the protection of individuals with regard to the automatic processing of personal data and on the free movement of such data⁸ and with Directive 2002/58/EC of 12 July 2002 concerning the processing of personal data and the protection of privacy⁹ in the electronic communications sector and in compliance with Article 7 of Regulation (EU) N^o. 165/2014.~~

(247) Each Master File (MF) of any tachograph card shall contain up to five Elementary Files (EF) for card management, application and chip identifications, and two Dedicated Files (DF):

- DF Tachograph, which contains the application accessible to first generation vehicle units, which is also present in first generation tachograph cards,
- DF Tachograph_G2, which contains the application only accessible to second generation vehicle units, which is only present in second generation tachograph cards.

Note: version 2 of second generation cards contains additional Elementary Files in DF Tachograph_G2.

The full details of the tachograph cards structure are specified in ~~Annex~~ **Sub-appendix 2**.

4.5.1 Elementary files for identification and card management

4.5.2 IC card identification

(248) Tachograph cards shall be able to store the following smart card identification data:

- clock stop,
- card serial number (including manufacturing references),
- card type approval number,
- card personaliser identification (ID),
- embedder ID,
- IC identifier.

4.5.2.1 Chip identification

⁸ OJ No L281, 23/11/1995, P 31

⁹ OJ No L201, 31/07/2002, P 37

(249) Tachograph cards shall be able to store the following Integrated Circuit (IC) identification data:

- IC serial number,
- IC manufacturing references.

4.5.2.2 DIR (only present in second generation tachograph cards)

(250) Tachograph cards shall be able to store the application identification data objects specified in ~~Appendix~~ **Sub-appendix 2**.

4.5.2.3 ATR information (conditional, only present in second generation tachograph cards)

(251) Tachograph cards shall be able to store the following extended length information data object:

in the case the tachograph card supports extended length fields, the extended length information data object specified in ~~Appendix~~ **Sub-appendix 2**.

4.5.2.4 Extended length information (conditional, only present in second generation tachograph cards)

(252) Tachograph cards shall be able to store the following extended length information data objects:

in the case the tachograph card supports extended length fields, the extended length information data objects specified in ~~Appendix~~ **Sub-appendix 2**.

4.5.3 Driver card

4.5.3.1 Tachograph application (accessible to first and second generation vehicle units)

4.5.3.1.1 Application identification

(253) The driver card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.3.1.2 Key and Certificates

(254) The driver card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix 11** part A.

4.5.3.1.3 Card identification

(255) The driver card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date.

4.5.3.1.4 Card holder identification

(256) The driver card shall be able to store the following card holder identification data:

- surname of the holder,
- first name(s) of the holder,
- date of birth,

- preferred language.

4.5.3.1.5 Card download

(257) The driver card shall be able to store the following data related to card download:

- date and time of last card download (for other purposes than control).

(258) The driver card shall be able to hold one such record.

4.5.3.1.6 Driving licence information

(259) The driver card shall be able to store the following driving licence data:

- issuing ~~Member States~~ **Contracting Party**, issuing authority name,
- driving licence number (at the date of the issue of the card).

4.5.3.1.7 Events data

For the purpose of this subparagraph, time shall be stored with a resolution of 1 second.

(260) The driver card shall be able to store data related to the following events detected by the ~~recording equipment~~ **control device** while the card was inserted:

- Time overlap (where this card is the cause of the event),
- Card insertion while driving (where this card is the subject of the event),
- Last card session not correctly closed (where this card is the subject of the event),
- Power supply interruption,
- Motion data error,
- Security breach attempts.

(261) The driver card shall be able to store the following data for these events:

- Event code,
- Date and time of beginning of the event (or of card insertion if the event was on-going at that time),
- Date and time of end of the event (or of card withdrawal if the event was on-going at that time),
- VRN and registering ~~Member State~~ **Contracting Party** of vehicle in which the event happened.

Note: For the “Time overlap” event:

- Date and time of beginning of the event shall correspond to the date and time of the card withdrawal from the previous vehicle,
- Date and time of end of the event shall correspond to the date and time of card insertion in current vehicle,
- Vehicle data shall correspond to the current vehicle raising the event.

Note: For the “Last card session not correctly closed” event:

- date and time of beginning of event shall correspond to the card insertion date and time of the session not correctly closed,

- date and time of end of event shall correspond to the card insertion date and time of the session during which the event was detected (current session),
- Vehicle data shall correspond to the vehicle in which the session was not correctly closed.

(262) The driver card shall be able to store data for the six most recent events of each type (i.e. 36 events).

4.5.3.1.8 Faults data

For the purpose of this subparagraph, time shall be recorded with a resolution of 1 second.

(263) The driver card shall be able to store data related to the following faults detected by the ~~recording equipment~~ **control device** while the card was inserted:

- Card fault (where this card is the subject of the ~~event~~ fault),
- ~~Recording equipment~~ fault
- **Control device fault.**

(264) The driver card shall be able to store the following data for these faults:

- Fault code,
- Date and time of beginning of the fault (or of card insertion if the fault was on-going at that time),
- Date and time of end of the fault (or of card withdrawal if the fault was on-going at that time),
- VRN and registering ~~Member States~~ **Contracting Party** of vehicle in which the fault happened.

(265) ~~The driver card shall be able to store data for the twelve most recent faults of each type (i.e. 24 faults)~~ **The driver card shall be able to store data for the twelve most recent faults of each type (i.e. 24 faults).**

4.5.3.1.9 Driver activity data

(266) The driver card shall be able to store, for each calendar day where the card has been used or for which the driver has entered activities manually, the following data:

- the date,
- a daily presence counter (increased by one for each of these calendar days),
- the total distance travelled by the driver during this day,
- a driver status at 00:00,
- whenever the driver has changed of activity, and/or has changed of driving status, and/or has inserted or withdrawn his card:
- the driving status (CREW, SINGLE),
- the slot (DRIVER, CO-DRIVER),
- the card status (INSERTED, NOT INSERTED),
- the activity (DRIVING, AVAILABILITY, WORK, BREAK/REST),
- the time of the change.

(267) The driver card memory shall be able to hold driver activity data for at least 28 days (the average activity of a driver is defined as 93 activity changes per day).

(268) The data listed under requirements 261, 264 and 266 shall be stored in a way allowing the retrieval of activities in the order of their occurrence, even in case of a time overlap situation.

4.5.3.1.10 Vehicles used data

(269) The driver card shall be able to store, for each calendar day where the card has been used, and for each period of use of a given vehicle that day (a period of use includes all consecutive insertion / withdrawal cycle of the card in the vehicle, as seen from the card point of view), the following data:

- date and time of first use of the vehicle (i.e. first card insertion for this period of use of the vehicle, or 00h00 if the period of use is on-going at that time),
- vehicle odometer value at that time,
- date and time of last use of the vehicle, (i.e. last card withdrawal for this period of use of the vehicle, or 23h59 if the period of use is on-going at that time),
- vehicle odometer value at that time,
- VRN and registering ~~Member State~~ **Contracting Party** of the vehicle.

(270) The driver card shall be able to store at least 84 such records.

4.5.3.1.11 Places where daily work periods start and/or end

(271) The driver card shall be able to store the following data related to places where daily work periods begin and/or end, entered by the driver:

- the date and time of the entry (or the date/time related to the entry if the entry is made during the manual entry procedure),
- the type of entry (begin or end, condition of entry),
- the country and region entered,
- the vehicle odometer value.

(272) The driver card memory shall be able to hold at least 42 pairs of such records.

4.5.3.1.12 Card session data

(273) The driver card shall be able to store data related to the vehicle which opened its current session:

- date and time the session was opened (i.e. card insertion) with a resolution of one second,
- VRN and registering ~~Member State~~ **Contracting Party**.

4.5.3.1.13 Control activity data

(274) The driver card shall be able to store the following data related to control activities:

- date and time of the control,
- control card number and card issuing ~~Member State~~ **Contracting Party**,
- type of the control (displaying and/or printing and/or VU downloading and/or card downloading (see note)),
- Period downloaded, in case of downloading,

- VRN and registering ~~Member State~~ **Contracting Party** of the vehicle in which the control happened.

Note: card downloading will only be recorded if performed through a ~~recording equipment~~ **control device**.

(275) The driver card shall be able to hold one such record.

4.5.3.1.14 Specific conditions data

(276) The driver card shall be able to store the following data related to specific conditions entered while the card was inserted (whatever the slot):

- Date and time of the entry,
- Type of specific condition.

(277) The driver card shall be able to store at least 56 such records.

4.5.3.2 Tachograph generation 2 application (not accessible to first generation vehicle units, **accessible to version 1 and version 2 of second generation vehicle units**)

4.5.3.2.1 Application identification

(278) The driver card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.3.2.1.1 Additional application identification (not accessed by version 1 of second generation vehicle units)

(278a) The driver card shall be able to store additional application identification data only applicable for version 2.

4.5.3.2.2 Keys and ~~Certificates~~ **certificates**

(279) The driver card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part B.

4.5.3.2.3 Card identification

(280) The driver card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date.

4.5.3.2.4 Card holder identification

(281) The driver card shall be able to store the following card holder identification data:

- surname of the holder,
- first name(s) of the holder,
- date of birth,
- preferred language.

4.5.3.2.5 Card download

(282) The driver card shall be able to store the following data related to card download:

date and time of last card download (for other purposes than control).

(283) The driver card shall be able to hold one such record.

4.5.3.2.6 Driving licence information

(284) The driver card shall be able to store the following driving licence data:

issuing ~~Member State~~ **Contracting Party**, issuing authority name,
driving licence number (at the date of the issue of the card).

4.5.3.2.7 Events data

For the purpose of this subparagraph, time shall be stored with a resolution of 1 second.

(285) The driver card shall be able to store data related to the following events detected by the ~~recording equipment~~ **control device** while the card was inserted:

- Time overlap (where this card is the cause of the event),
- Card insertion while driving (where this card is the subject of the event),
- Last card session not correctly closed (where this card is the subject of the event),
- Power supply interruption,
- Communication error with the remote communication facility,
- Absence of position information from GNSS receiver event,
- Communication error with the external GNSS facility
- Motion data error,
- Vehicle motion conflict,
- Security breach attempts,
- Time conflict.

(286) The driver card shall be able to store the following data for these events:

- Event code,
- Date and time of beginning of the event (or of card insertion if the event was on-going at that time),
- Date and time of end of the event (or of card withdrawal if the event was on-going at that time),
- VRN and registering ~~Member State~~ **Contracting Party** of vehicle in which the event happened.

Note: For the “Time overlap” event:

- Date and time of beginning of the event shall correspond to the date and time of the card withdrawal from the previous vehicle,
- Date and time of end of the event shall correspond to the date and time of card insertion in current vehicle,
- Vehicle data shall correspond to the current vehicle raising the event.

Note: For the “Last card session not correctly closed” event:

- date and time of beginning of event shall correspond to the card insertion date and time of the session not correctly closed,

- date and time of end of event shall correspond to the card insertion date and time of the session during which the event was detected (current session),
- Vehicle data shall correspond to the vehicle in which the session was not correctly closed.

(287) The driver card shall be able to store data for the ~~six~~**12** most recent events of each type (i.e. ~~66~~**132** events).

4.5.3.2.8 Faults data

For the purpose of this subparagraph, time shall be recorded with a resolution of 1 second.

(288) The driver card shall be able to store data related to the following faults detected by the ~~recording equipment~~ **control device** while the card was inserted:

- Card fault (where this card is the subject of the ~~event recording equipment fault~~),
- **Control device fault.**

(289) The driver card shall be able to store the following data for these faults:

- Fault code,
- Date and time of beginning of the fault (or of card insertion if the fault was on-going at that time),
- Date and time of end of the fault (or of card withdrawal if the fault was on-going at that time),
- VRN and registering ~~Member State~~ **Contracting Party** of vehicle in which the fault happened.

(290) The driver card shall be able to store data for the ~~24~~**twelve** most recent faults of each type (i.e. ~~24~~**48** faults). ~~The driver card shall be able to store data for the twelve most recent faults of each type (i.e. 24 faults)~~

4.5.3.2.9 Driver activity data

(291) The driver card shall be able to store, for each calendar day where the card has been used or for which the driver has entered activities manually, the following data:

- the date,
- a daily presence counter (increased by one for each of these calendar days),
- the total distance travelled by the driver during this day,
- a driver status at 00:00,
 - whenever the driver has changed of activity, and/or has changed of driving status, and/or has inserted or withdrawn his card:
 - the driving status (CREW, SINGLE)
 - the slot (DRIVER, CO-DRIVER),
 - the card status (INSERTED, NOT INSERTED),
 - the activity (DRIVING, AVAILABILITY, WORK, BREAK/REST).
 - the time of the change,

(292) The driver card memory shall be able to hold driver activity data for ~~at least 28~~**56** days (the average activity of a driver is defined **for this requirement** as ~~93~~**117** activity changes per day).

(293) The data listed under requirements 286, 289 and 291 shall be stored in a way allowing the retrieval of activities in the order of their occurrence, even in case of a time overlap situation.

4.5.3.2.10 Vehicles used data

(294) The driver card shall be able to store, for each calendar day where the card has been used, and for each period of use of a given vehicle that day (a period of use includes all consecutive insertion/withdrawal cycle of the card in the vehicle, as seen from the card point of view), the following data:

- date and time of first use of the vehicle (i.e. first card insertion for this period of use of the vehicle, or 00h00 if the period of use is on-going at that time),
- vehicle odometer value at that first use time,
- date and time of last use of the vehicle, (i.e. last card withdrawal for this period of use of the vehicle, or 23h59 if the period of use is on-going at that time),
- vehicle odometer value at that last use time,
- VRN and registering ~~Member State~~ **Contracting Party** of the vehicle,
- VIN of the vehicle.

(295) The driver card shall be able to store ~~200~~ **at least 84** such records.

4.5.3.2.11 Places and positions where daily work periods start and/or end

(296) The driver card shall be able to store the following data related to places where daily work periods begin and/or end, entered by the driver:

the date and time of the entry (or the date/time related to the entry if the entry is made during the manual entry procedure),

- the type of entry (begin or end, condition of entry),
- the country and region entered,
- the vehicle odometer value,
- the vehicle position,
- the GNSS accuracy, date and time when the position was determined.

(297) The driver card memory shall be able to hold ~~at least 11284 pairs of~~ such records.

4.5.3.2.12 Card session data

(298) The driver card shall be able to store data related to the vehicle which opened its current session:

- date and time the session was opened (i.e. card insertion) with a resolution of one second,
- VRN and registering ~~Member State~~ **Contracting Party**.

4.5.3.2.13 Control activity data

(299) The driver card shall be able to store the following data related to control activities:

date and time of the control,

control card number and card issuing Contracting Party,

type of the control (displaying and/or printing and/or VU downloading and/or card downloading (see note)),

Period downloaded, in case of downloading,

VRN and registering ~~Member State~~ **Contracting Party** of the vehicle in which the control happened.

Note: security requirements imply that card downloading will only be recorded if performed through a ~~recording equipment~~ **control device**.

(300) The driver card shall be able to hold one such record.

4.5.3.2.14 Specific conditions data

(301) The driver card shall be able to store the following data related to specific conditions entered while the card was inserted (whatever the slot):

- Date and time of the entry,
- Type of specific condition.

(302) The driver card shall be able to store ~~112 at least 56~~ such records.

4.5.3.2.15 Vehicle units used data

(303) The driver card shall be able to store the following data related to the different vehicle units in which the card was used:

- the date and time of the beginning of the period of use of the vehicle unit (i.e. first card insertion in the vehicle unit for the period),
- the manufacturer of the vehicle unit,
- the vehicle unit type,
- the vehicle unit software version number.

(304) The driver card shall be able to store ~~200 at least 84~~ such records.

4.5.3.2.16 Three hours-~~continuous~~ **accumulated** driving places data

(305) The driver card shall be able to store the following data related to the position of the vehicle where the ~~continuous~~ **accumulated** driving time of ~~the driver~~ reaches a multiple of three hours:

- the date and time when the ~~continuous~~ **accumulated** driving time of ~~the card holder~~ reaches a multiple of three hours,
- the position of the vehicle,
- the GNSS accuracy, date and time when the position was determined,
- **the vehicle odometer value.**

(306) The driver card shall be able to store ~~336 at least 252~~ such records.

4.5.3.2.17 **Authentication status for positions related to places where daily work periods start and/or end (not accessed by version 1 of second generation vehicle units)**

(306a) The driver card shall be able to store additional data related to places where daily work periods begin and/or end, entered by the driver in accordance with point 4.5.3.2.11:

- **the date and time of the entry, which shall be exactly the same date and time as the one stored in EF Places under the DF Tachograph_G2,**

- a flag indicating whether the position has been authenticated.

(306b) The driver card memory shall be able to hold 112 such records.

4.5.3.2.18 Authentication status for positions where three hours accumulated driving time are reached (not accessed by version 1 of second generation vehicle units)

(306c) The driver card shall be able to store additional data related to the position of the vehicle where the accumulated driving time reaches a multiple of three hours in accordance with point 4.5.3.2.16:

- the date and time when the accumulated driving time reaches a multiple of three hours, which shall be exactly the same date and time as the one stored in EF GNSS_Places under the DF Tachograph_G2,
- a flag indicating whether the position has been authenticated.

(306d) The driver card shall be able to store 336 such records.

4.5.3.2.19 Border crossings (not accessed by version 1 of second generation vehicle units)

(306e) The driver card shall be able to store the following data related to border crossings either upon card insertion in accordance with requirement 147b or with the card already inserted:

- the country that the vehicle is leaving,
- the country that the vehicle is entering,
- the date and time when the vehicle has crossed the border,
- the position of the vehicle when the border was crossed,
- the GNSS accuracy,
- a flag indicating whether the position has been authenticated,
- the vehicle odometer value.

(306f) The driver card memory shall be able to store 1120 such records.

4.5.3.2.20 Load/unload operations (not accessed by version 1 of second generation vehicle units)

(306g) The driver card shall be able to store the following data related to load/unload operations:

- operation type (load, unload or simultaneous load/unload),
- the date and time of the load/unload operation,
- the position of the vehicle,
- the GNSS accuracy, date and time when the position was determined,
- a flag indicating whether the position has been authenticated,
- the vehicle odometer value.

(306h) The driver card shall be able to store 1624 load/unload operations.

4.5.3.2.21 Load type entries (not accessed by version 1 of second generation vehicle units)

(306i) The driver card shall be able to store the following data related to load type automatically entered by the VU at each card insertion:

- the load type entered (goods or passengers),
- the date and time of the entry.

(306j) The driver card shall be able to store 336 such records.

4.5.3.2.22 VU configurations (not accessed by version 1 of second generation vehicle units)

(306k) The driver card shall be able to store the card owner tachograph specific settings.

(306l) The driver card storage capacity for card owner tachograph specific settings shall be 3072 bytes.

4.5.4 Workshop card

4.5.4.1 Tachograph application (accessible to first and second generation vehicle units)

4.5.4.1.1 Application identification

(307) The workshop card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.4.1.2 Keys and ~~Certificates~~—certificates

(308) The workshop card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part A.

309) The workshop card shall be able to store a Personal Identification Number (PIN code).

4.5.4.1.3 Card identification

(310) The workshop card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date.

4.5.4.1.4 Card holder identification

(311) The workshop card shall be able to store the following card holder identification data:

- workshop name,
- workshop address,
- surname of the holder,
- first name(s) of the holder,
- preferred language.

4.5.4.1.5 Card download

(312) The workshop card shall be able to store a card download data record in the same manner as a driver card.

4.5.4.1.6 Calibration and time adjustment data

(313) The workshop card shall be able to hold records of calibrations and/or time adjustments performed while the card is inserted in a ~~recording equipment~~ **control device**.

(314) Each calibration record shall be able to hold the following data:

- Purpose of calibration (activation, first installation, installation, periodic inspection,),
- Vehicle identification,
- Parameters updated or confirmed (w, k, l, tyre size, speed limiting device setting, odometer (new and old values), date and time (new and old values),
- ~~Recording equipment~~ **Control device** identification (VU part number, VU serial number, motion sensor serial number).

(315) The workshop card shall be able to store at least 88 such records.

(316) The workshop card shall hold a counter indicating the total number of calibrations performed with the card.

(317) The workshop card shall hold a counter indicating the number of calibrations performed since its last download.

4.5.4.1.7 Events and faults data

(318) The workshop card shall be able to store events and faults data records in the same manner as a driver card.

(319) The workshop card shall be able to store data for the three most recent events of each type (i.e. 18 events) and the six most recent faults of each type (i.e. 12 faults).

4.5.4.1.8 Driver activity data

(320) The workshop card shall be able to store driver activity data in the same manner as a driver card.

(321) The workshop card shall be able to hold driver activity data for at least 1 day of average driver activity.

4.5.4.1.9 Vehicles used data

(322) The workshop card shall be able to store vehicles used data records in the same manner as a driver card.

(323) The workshop card shall be able to store at least 4 such records.

4.5.4.1.10 Daily work periods start and/or end data

(324) The workshop card shall be able to store daily works period start and/or end data records in the same manner as a driver card.

(325) The workshop card shall be able to hold at least 3 pairs of such records.

4.5.4.1.11 Card session data

(326) The workshop card shall be able to store a card session data record in the same manner as a driver card.

4.5.4.1.12 Control activity data

(327) The workshop card shall be able to store a control activity data record in the same manner as a driver card.

4.5.4.1.13 Specific conditions data

(328) The workshop card shall be able to store data relevant to specific conditions in the same manner as the driver card.

(329) The workshop card shall be able to store at least 2 such records.

4.5.4.2 Tachograph Generation 2 application (not accessible to first generation vehicle units, **accessible to version 1 and version 2 of second generation vehicle units**)

4.5.4.2.1 Application identification

(330) The workshop card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.4.2.1.1 Additional application identification (not accessed by version 1 of second generation vehicle units)

(330a) The workshop card shall be able to store additional application identification data only applicable for version 2.

4.5.4.2.2 Keys and ~~Certificates~~ **certificates**

(331) The workshop card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part B.

(332) The workshop card shall be able to store a Personal Identification Number (PIN code).

4.5.4.2.3 Card identification

(333) The workshop card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date.

4.5.4.2.4 Card holder identification

(334) The workshop card shall be able to store the following card holder identification data:

- workshop name,
- workshop address,
- surname of the holder,
- first name(s) of the holder,
- preferred language.

4.5.4.2.5 Card download

(335) The workshop card shall be able to store a card download data record in the same manner as a driver card.

4.5.4.2.6 Calibration and time adjustment data

(336) The workshop card shall be able to hold records of calibrations and/or time adjustments performed while the card is inserted in a ~~recording equipment~~ **control device**.

- (337) Each calibration record shall be able to hold the following data:
- purpose of calibration (activation, first installation, installation, periodic inspection,)
 - vehicle identification,
 - parameters updated or confirmed (w, k, l, tyre size, speed limiting device setting, odometer (new and old values), date and time (new and old values),
 - ~~recording equipment~~ **control device** identification (VU part number, VU serial number, motion sensor serial number, remote communication facility serial number and external GNSS facility serial number, if applicable),
 - seal type and identifier of all seals in place,
 - ability of the VU to use first generation tachograph cards (enabled or not).
- (338) The workshop card shall be able to store ~~255 at least 88~~ such records.
- (339) The workshop card shall hold a counter indicating the total number of calibrations performed with the card.
- (340) The workshop card shall hold a counter indicating the number of calibrations performed since its last download.
- 4.5.4.2.7 Events and faults data
- (341) The workshop card shall be able to store events and faults data records in the same manner as a driver card.
- (342) The workshop card shall be able to store data for the three most recent events of each type (i.e. 33 events) and the six most recent faults of each type (i.e. 12 faults).
- 4.5.4.2.8 Driver activity data
- (343) The workshop card shall be able to store driver activity data in the same manner as a driver card.
- (344) The workshop card shall be able to hold driver activity data for ~~at least 1~~ day **containing 240 activity changes** of average driver activity.
- 4.5.4.2.9 Vehicles used data
- (345) The workshop card shall be able to store vehicles used data records in the same manner as a driver card.
- (346) The workshop card shall be able to store ~~8 at least 4~~ such records.
- 4.5.4.2.10 **Places and positions where** ~~D~~daily work periods start and/or end data
- (347) The workshop card shall be able to store **places and positions where** daily work periods ~~start~~ **begin** and/or end data records in the same manner as a driver card.
- (348) The workshop card shall be able to ~~store 4~~ **hold at least 3** pairs of such records.
- 4.5.4.2.11 Card session data
- (349) The workshop card shall be able to store a card session data record in the same manner as a driver card.
- 4.5.4.2.12 Control activity data
- (350) The workshop card shall be able to store a control activity data record in the same manner as a driver card.
- 4.5.4.2.13 Vehicle units used data

(351) The workshop card shall be able to store the following data related to the different vehicle units in which the card was used:

- the date and time of the beginning of the period of use of the vehicle unit (i.e. first card insertion in the vehicle unit for the period),
- the manufacturer of the vehicle unit,
- the vehicle unit type,
- the vehicle unit software version number.

(352) The workshop card shall be able to store ~~8~~**at least 4** such records.

4.5.4.2.14 ~~Three hours continuous~~**accumulated** driving places data

(353) The workshop card shall be able to store the following data related to the position of the vehicle where the ~~continuous~~**accumulated** driving time of the driver reaches a multiple of three hours:

- the date and time when the ~~continuous~~**accumulated** driving time of the card holder reaches a multiple of three hours,
- the position of the vehicle,
- the GNSS accuracy, date and time when the position was determined,
- **the vehicle odometer value.**

(354) The workshop card shall be able to store ~~24~~**at least 18** such records.

4.5.4.2.15 Specific conditions data

(355) The workshop card shall be able to store data relevant to specific conditions in the same manner as the driver card.

(356) The workshop card shall be able to store at least ~~42~~ such records.

4.5.4.2.16 Authentication status for positions related to places where daily work periods start and/or end (not accessed by version 1 of second generation vehicle units)

(356a) The workshop card shall be able to store additional data related to places where daily work periods start and/or end in the same manner as a driver card.

(356b) The workshop card memory shall be able to store 4 pairs of such records.

4.5.4.2.17 Authentication status for positions where three hours accumulated driving are reached (not accessed by version 1 of second generation vehicle units)

(356c) The workshop card shall be able to store additional data related to the position of the vehicle where the accumulated driving time reaches a multiple of three hours in the same manner as a driver card.

(356d) The workshop card shall be able to store at least 24 such records.

4.5.4.2.18 Border crossings (not accessed by version 1 of second generation vehicle units)

(356e) The workshop card shall be able to store the border crossings in the same manner as a driver card.

(356f) The workshop card memory shall be able to store 4 such records.

4.5.4.2.19 Load/unload operations (not accessed by version 1 of second generation vehicle units)

(356g) The workshop card shall be able to store the load/unload operations in the same manner as a driver card.

(356h) The workshop card shall be able to store 8 load, unload or simultaneous load/unload operations.

4.5.4.2.20 Load type entries (not accessed by version 1 of second generation vehicle units)

(356i) The workshop card shall be able to store the load type entries in the same manner as a driver card.

(356j) The workshop card shall be able to store 4 such records.

4.5.4.2.21 Calibration Additional Data (not accessed by version 1 of second generation vehicle units)

(356k) The workshop card shall be able to store additional calibration data only applicable for version 2:

- the old date and time and the vehicle identification number, which shall be exactly the same values as the one stored in EF Calibration under the DF Tachograph_G2,
- the by-default load type entered during this calibration,
- the country in which the calibration has been performed, and the date time when the position used to determine this country was provided by the GNSS receiver.

(356l) The workshop card shall be able to store 255 such records.

4.5.4.2.22 VU configurations (not accessed by version 1 of second generation vehicle units)

(356m) The workshop card shall be able to store the card owner tachograph specific settings.

(356n) The workshop card storage capacity for card owner tachograph specific settings shall be 3072 bytes.

4.5.5 Control card

4.5.5.1 Tachograph application (accessible to first and second generation vehicle units)

4.5.5.1.1 Application identification

(357) The control card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.5.1.2 Keys and Certificates

(358) The control card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part A.

4.5.5.1.3 Card identification

(359) The control card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,

- card beginning of validity date, card expiry date (if any).

4.5.5.1.4 Card holder identification

(360) The control card shall be able to store the following card holder identification data:

- control body name,
- control body address,
- surname of the holder,
- first name(s) of the holder,
- preferred language.

4.5.5.1.5 Control activity data

(361) The control card shall be able to store the following control activity data:

- date and time of the control,
- type of the control (displaying and/or printing and/or VU downloading and/or card downloading ~~and/or roadside calibration checking~~),
- period downloaded (if any),
- VRN and ~~Member State~~ **Contracting Party** registering authority of the controlled vehicle,
- card number and card issuing ~~Member State~~ **Contracting Party** of the driver card controlled.

(362) The control card shall be able to hold at least 230 such records.

4.5.5.2 Tachograph G2 application (not accessible to first generation vehicle unit)

4.5.5.2.1 Application identification

(363) The control card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.5.2.1.1 Additional application identification (not accessed by version 1 of second generation vehicle units)

(363a) The control card shall be able to store additional application identification data only applicable for version 2.

4.5.5.2.2 Keys and ~~Certificates~~ **certificates**

(364) The control card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part B.

4.5.5.2.3 Card identification

(365) The control card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date (if any).

4.5.5.2.4 Card holder identification

(366) The control card shall be able to store the following card holder identification data:

- control body name,
- control body address,
- surname of the holder,
- ~~F~~first name(s) of the holder,
- preferred language.

4.5.5.2.5 Control activity data

(367) The control card shall be able to store the following control activity data:

- date and time of the control,
- type of the control (displaying and/or printing and/or VU downloading and/or card downloading and/or roadside calibration checking)
- period downloaded (if any),
- VRN and ~~Member State~~ **Contracting Party** registering authority of the controlled vehicle,
- card number and card issuing ~~Member State~~ **Contracting Party** of the driver card controlled.

(368) The control card shall be able to hold at least 230 such records.

4.5.5.2.6 VU configurations (not accessed by version 1 of second generation vehicle units)

(368a) The control card shall be able to store the card owner tachograph specific settings.

(368b) The control card storage capacity for card owner tachograph specific settings shall be 3072 bytes.

4.5.6 Company card

4.5.6.1 Tachograph application (accessible to first and second generation vehicle units)

4.5.6.1.1 Application identification

(369) The company card shall be able to store the following application identification data:

- tachograph application identification,
- type of tachograph card identification.

4.5.6.1.2 Keys and ~~Certificates~~ **certificates**

(370) The company card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part A.

4.5.6.1.3 Card identification

(371) The company card shall be able to store the following card identification data:

- card number,
- issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
- card beginning of validity date, card expiry date (if any).

4.5.6.1.4 Card holder identification

- (372) The company card shall be able to store the following card holder identification data:
- company name,
 - company address.

4.5.6.1.5 Company activity data

- (373) The company card shall be able to store the following company activity data:
- date and time of the activity,
 - type of the activity (VU locking in and/or out, and/or VU downloading and/or card downloading)
 - period downloaded (if any),
 - VRN and ~~Member State~~ **Contracting Party** registering authority of vehicle,
 - card number and card issuing ~~Member State~~ **Contracting Party** (in case of card downloading).

- (374) The company card shall be able to hold at least 230 such records.

4.5.6.2 Tachograph G2 application (not accessible to first generation vehicle unit)

4.5.6.2.1 Application identification

- (375) The company card shall be able to store the following application identification data:
- tachograph application identification,
 - type of tachograph card identification.

4.5.6.2.1.1 Additional application identification (not accessed by version 1 of second generation vehicle units)

- (375a) The company card shall be able to store additional application identification data only applicable for version 2.**

4.5.6.2.2 Keys and ~~Certificates~~ **certificates**

- (376) The company card shall be able to store a number of cryptographic keys and certificates, as specified in ~~Appendix~~ **Sub-appendix** 11 part B.

4.5.6.2.3 Card identification

- (377) The company card shall be able to store the following card identification data:
- card number,
 - issuing ~~Member State~~ **Contracting Party**, issuing authority name, issue date,
 - card beginning of validity date, card expiry date (if any).

4.5.6.2.4 Card holder identification

- (378) The company card shall be able to store the following card holder identification data:
- company name,
 - company address.

4.5.6.2.5 Company activity data

- (379) The company card shall be able to store the following company activity data:

- date and time of the activity,
- type of the activity (VU locking in and/or out, and/or VU downloading and/or card downloading)
- period downloaded (if any),
- VRN and ~~Member State~~ **Contracting Party** registering authority of vehicle,
- card number and card issuing ~~Member State~~ **Contracting Party** (in case of card downloading).

(380) The company card shall be able to hold at least 230 such records.

4.5.6.2.6 VU configurations (not accessed by version 1 of second generation vehicle units)

(380a) The company card shall be able to store the card owner tachograph specific settings.

(380b) The company card storage capacity for card owner tachograph specific settings shall be 3072 bytes.

5. Installation of the ~~recording equipment~~ control device

5.1. Installation

(381) New ~~recording equipment~~ **control device** shall be delivered non-activated to fitters or vehicle manufacturers, with all calibration parameters, as listed in Chapter 3.21, set to appropriate and valid default values. Where no particular value is appropriate, literal parameters shall be set to strings of “?” and numeric parameters shall be set to “0”. Delivery of security relevant parts of the ~~recording equipment~~ **control device** can be restricted if required during security certification.

(382) Before its activation, the ~~recording equipment~~ **control device** shall give access to the calibration function even if not in calibration mode.

(383) Before its activation, the ~~recording equipment~~ **control device** shall neither record nor store data referred to by **the requirements 102 to 133** ~~points 3.12.3, 3.12.9 and 3.12.12 to 3.12.15~~ inclusive. **Nevertheless, before its activation, the recording equipment may record and store the security breach attempt events in accordance with requirement 117, and the recording equipment faults in accordance with requirement 118.**

(384) During installation, vehicle manufacturers shall pre-set all known parameters.

(385) Vehicle manufacturers or fitters shall activate the installed ~~recording equipment~~ **control device** at the latest before the vehicle is used in scope ~~Regulation (EC) N° 561/2006~~ of this **Agreement**.

(386) The activation of the ~~recording equipment~~ **control device** shall be triggered automatically by the first insertion of a valid workshop card in either of its card interface devices.

(387) Specific pairing operations required between the motion sensor and the vehicle unit, if any, shall take place automatically before or during activation.

(388) In a similar way, specific coupling operations between the external GNSS facility and the vehicle unit, if any, shall take place automatically before or during activation.

(389) After its activation, the ~~recording equipment~~ **control device** shall fully enforce functions and data access rights.

(390) After its activation, the ~~recording equipment~~ **control device** shall communicate to the remote communication facility the secured data necessary for the purpose of targeted roadside checks.

(391) The recording and storing functions of the ~~recording equipment~~ **control device** shall be fully operational after its activation.

(392) Installation shall be followed by a calibration. The first calibration may not necessarily include entry of the vehicle registration ~~identification number~~ (**VRN and Contracting Party**), when it is not known by the approved workshop having to undertake this calibration. In these circumstances, it shall be possible, for the vehicle owner, and at this time only, to enter the **VRN and the Contracting Party** using his ~~company card~~ prior to using the vehicle in scope of ~~Regulation (EC) N° 561/2006~~ **this Agreement** (e.g. by using commands through an appropriate menu structure of the vehicle unit's man-machine interface⁴⁰). Any update or confirmation of this entry shall only be possible using a Workshop Card.

(393) The installation of an external GNSS facility requires the coupling with the vehicle unit and the subsequent verification of the GNSS position information.

(394) The ~~recording equipment~~ **control device** must be positioned in the vehicle in such a way as to allow the driver to access the necessary functions from his seat.

5.2. Installation plaque

(395) After the ~~recording equipment~~ **control device** has been checked on installation, an installation plaque, engraved or printed in a permanent way, which is clearly visible and easily accessible shall be affixed onto the control device. In cases where this is not possible, the plaque shall be affixed to the vehicle's "B" pillar so that it is clearly visible. For vehicles that do not have a "B" pillar, the installation plaque should be affixed ~~in the area of the door frame on the driver's side~~ **in the area of the** door frame on the driver's side of the vehicle and be clearly visible in all cases.

After every inspection by an approved fitter or workshop, a new plaque shall be affixed in place of the previous one.

(396) The plaque shall bear at least the following details:

- name, address or trade name of the approved fitter or workshop,
- characteristic coefficient of the vehicle, in the form "w = ... imp/km",
- constant of the ~~recording equipment~~ **control device**, in the form "k = ... imp/km",
- effective circumference of the wheel tyres in the form "l = ... mm",
- tyre size,
- the date on which the characteristic coefficient of the vehicle and the effective circumference of the wheel tyres were measured,
- the vehicle identification number,
- the presence (or not) of an external GNSS facility,
- the serial number of the external GNSS facility, **if applicable**,
- the serial number of the remote communication ~~facility device~~ **facility device, if any**,

⁴⁰ OJ L 11.4.2006, p1

- the serial number of all the seals in place,
- the part of the vehicle where the adaptor, if any, is installed,
- the part of the vehicle where the motion sensor is installed, if not connected to the gear-box or an adaptor is not being used,
- a description of the colour of the cable between the adaptor and that part of the vehicle providing its incoming impulses,
- the serial number of the embedded motion sensor of the adaptor,
- **the by-default load type associated to the vehicle.**

(397) For M1 and N1 vehicles only, and which are fitted with an adaptor in conformity with Regulation (EC) N^o. 68/2009¹¹ **Sub-appendix 16** as last amended and where it is not possible to include all the information necessary, as described in ~~Requirement~~ **requirement 396**, a second, additional, plaque may be used. In such cases, this additional plaque shall contain at least the last four indents described in ~~Requirement~~ **requirement 396**.

This second, additional plaque, if used, shall be affixed next to or beside the first primary plaque described in ~~Requirement~~ **requirement 396**, and shall have the same protection level. Furthermore, the secondary plaque shall also bear the name, address or trade name of the approved fitter or workshop that carried out the installation, and the date of installation.

5.3. Sealing

(398) The following parts shall be sealed:

- Any connection which, if disconnected, would cause undetectable alterations to be made or undetectable data loss (this may e.g. apply for the motion sensor fitting on the gearbox, the adaptor for M1/N1 vehicles, the external GNSS connection or the vehicle unit);
- The installation plaque, unless it is attached in such a way that it cannot be removed without the markings thereon being destroyed.

(398a) The seals mentioned above shall be certified according to the standard EN 16882:2016.¹²

(399) The seals mentioned above may be removed:

- In case of emergency,
- To install, to adjust or to repair a speed limitation device or any other device contributing to road safety, provided that the ~~recording equipment~~ **control device** continues to function reliably and correctly and is resealed by an approved fitter or workshop (in accordance with Chapter 6) immediately after fitting the speed limitation device or any other device contributing to road safety or within seven days in other cases.

(400) On each occasion that these seals are broken a written statement giving the reasons for such action shall be prepared and made available to the competent authority.

(401) Seals shall hold an identification number, allocated by its manufacturer. This number shall be unique and distinct from any other seal number allocated by any other seals manufacturer.

¹¹ OJL 21.1.2009, p3

¹² **Conversion to ISO standard planned over a five-year period**

This unique identification number is defined as: ~~MM NNNNN~~ **MMNNNNNNNN** by non-removable marking, with MM as unique manufacturer identification (database registration to be managed by EC) and ~~NNNN NNNNNNNN~~ seal alpha-numeric number, unique in the manufacturer domain.

(402) The seals shall have a free space where approved fitters, workshops or vehicle manufacturers can add a special mark ~~according to Article 22.3 of the Regulation (EU) no 165/2014.~~

This mark shall not cover the seal identification number.

(403) Seals manufacturers shall be registered in a dedicated database **when they get a seal model certified according to EN 16882:2016** and shall make their identification seals numbers public ~~through a procedure to be established by the European Commission.~~

(404) Approved workshops and vehicle manufacturers shall, in the frame of ~~Regulation (EU) No. 165/2014~~ **this Agreement**, only use ~~seals~~ **certified seals according to EN 16882:2016** from those of the seals manufacturers listed in the data base mentioned above.

(405) Seal manufacturers and their distributors shall maintain full traceability records of the seals sold to be used in the frame of ~~Regulation (EU) N° 165/2014~~ **this Agreement** and shall be prepared to produce them to competent national authorities whenever need be.

(406) Seals unique identification numbers shall be visible on the installation plaque.

6. Checks, inspections and repairs

Requirements on the circumstances in which seals may be removed, ~~as referred to in Article 22.5 of Regulation (EU) No 165/2014~~ are defined in Chapter 5.3 of this ~~annex~~ **Appendix**.

6.1. Approval of fitters, workshops and vehicle manufacturers

The ~~Member States~~ **Contracting Parties** approve, regularly control and certify the bodies to carry out:

- installations,
- checks,
- inspections,
- repairs.

Workshop cards shall be issued only to fitters and/or workshops approved for the activation and/or the calibration of ~~recording equipment~~ **control device** in conformity with this ~~annex~~ **Appendix** and, unless duly justified:

- who are not eligible for a company card;
- and whose other professional activities do not present a potential compromise of the overall security of the system as required in ~~Appendix~~ **Sub-appendix 10**.

6.2. Check of new or repaired ~~instruments~~ components

(407) Every individual device, whether new or repaired, shall be checked in respect of its proper operation and the accuracy of its reading and recordings, within the limits laid down in Chapter 3.2.1, 3.2.2, 3.2.3 and 3.3 ~~by means of sealing in accordance with Chapter 5.3 and calibration.~~

6.3. Installation inspection

(408) When being fitted to a vehicle, the whole installation (including the ~~recording equipment~~ **control device**) shall comply with the provisions relating to maximum tolerances laid down in Chapter 3.2.1, 3.2.2, 3.2.3 and 3.3. **The whole installation shall be sealed in accordance with Chapter 5.3 and it shall include a calibration.**

6.4. Periodic inspections

(409) Periodic inspections of the equipment fitted to the vehicles shall take place after any repair of the equipment, or after any alteration of the characteristic coefficient of the vehicle or of the effective circumference of the tyres, or after equipment UTC time is wrong by more than ~~520~~ minutes, or when the VRN has changed, and at least once within two years (24 months) of the last inspection.

(410) These inspections shall include the following checks:

- that the ~~recording equipment~~ **control device** is working properly, including the data storage in tachograph cards function and the communication with remote communication readers,
- that compliance with the provisions of chapter 3.2.1 and 3.2.2 on the maximum tolerances on installation is ensured,
- that compliance with the provisions of chapter 3.2.3 and 3.3 is ensured,
- that the ~~recording equipment~~ **control device** carries the type approval mark,
- that the installation plaque, as defined by ~~Requirement~~ **requirement 396**, and the descriptive plaque, as defined by ~~Requirement~~ **requirement 225**, are affixed,
- the tyre size and the actual circumference of the tyres,
- that there are no manipulation devices attached to the equipment,
- that seals are correctly placed, in good state, that their identification numbers are valid (referenced seal manufacturer in the EC database) and that their identification numbers correspond to the installation plaque markings (see requirement 401),
- **that the version identifier of the stored digital map is the most recent one.**

(410a) In case of detection of a manipulation by the competent national authorities, the vehicle may be sent to an authorised workshop for a recalibration of the recording equipment.

(411) If one of the events listed in Chapter 3.9 (Detection of Events and/or Faults) is found to have occurred since the last inspection and is considered by tachograph manufacturers and/or national authorities as potentially putting the security of the equipment at risk, the workshop shall:

- (a) make a comparison between the motion sensor identification data of the motion sensor plugged into the gearbox with that of the paired motion sensor registered in the vehicle unit;
- (b) check if the information recorded on the installation plaque matches with the information contained within the vehicle unit record;
- (c) check if the motion sensor serial number and approval number, if printed on the body of the motion sensor, matches the information stored in the ~~recording equipment~~ **control device** data memory;

(d) compare identification data marked on the descriptive plaque of the external GNSS facility, if any, to the ones stored in the vehicle unit data memory;

(412) Workshops shall keep traces in their inspection reports of any findings concerning broken seals or manipulations devices. These reports shall be kept by workshops for at least 2 years and made available to the Competent Authority whenever requested to do so.

(413) These inspections shall include a calibration and a preventive replacement of the seals whose fitting is under the responsibility of workshops.

6.5. Measurement of errors

(414) The measurement of errors on installation and during use shall be carried out under the following conditions, which are to be regarded as constituting standard test conditions:

- vehicle unladen, in normal running order,
- tyre pressures in accordance with the manufacturer's instructions,
- tyre wear, within the limits allowed by national law,
- vehicle movement:
- the vehicle shall advance under its own engine power in a straight line on level ground and at a speed of 50 ± 5 km/h. The measuring distance shall be at least 1000m.
- provided that it is of comparable accuracy, alternative methods, such as a suitable test bench, may also be used for the test.

6.6. Repairs

415) Workshops shall be able to download data from the ~~recording equipment~~ **control device** to give the data back to the appropriate transport company.

416) Approved workshops shall issue to transport companies a certificate of data un-downloadability where the malfunction of the ~~recording equipment~~ **control device** prevents previously recorded data to be downloaded, even after repair by this workshop. The workshops will keep a copy of each issued certificate for at least two years.

7. ~~Card-issuing~~ issuing

The card issuing processes set-up by the ~~Member States~~ **Contracting Parties** shall conform to the following:

(417) The card number of the first issue of a tachograph card to an applicant shall have a consecutive index (if applicable) and a replacement index and a renewal index set to "0".

(418) The card numbers of all non-personal tachograph cards issued to a single control body or a single workshop or a single transport company shall have the same first 13 digits, and shall all have a different consecutive index.

(419) A tachograph card issued in replacement of an existing tachograph card shall have the same card number than the replaced one except the replacement index which shall be raised by "1" (in the order 0, ..., 9, A, ..., Z).

(420) A tachograph card issued in replacement of an existing tachograph card shall have the same card expiry date as the replaced one.

(421) A tachograph card issued in renewal of an existing tachograph card shall have the same card number as the renewed one except the replacement index which shall be reset to “0” and the renewal index which shall be raised by “1” (in the order 0, ..., 9, A, ..., Z).

(422) The exchange of an existing tachograph card, in order to modify administrative data, shall follow the rules of the renewal if within the same ~~Member States Contracting Party~~, or the rules of a first issue if performed by another ~~Member States Contracting Party~~.

(423) The “card holder surname” for non-personal workshop or control cards shall be filled with workshop or control body name or with the fitter or control officer’s name would ~~Member States Contracting Parties~~ so decide.

~~Member States shall exchange data electronically in order to ensure the uniqueness of driver cards that they issue in accordance with Article 31 of Regulation (EU) N° 165/2014.~~

(424) Reserved.

8. Type approval of ~~recording equipment~~ control devices and tachograph cards

8.1. General points

For the purpose of this chapter, the words ~~recording equipment~~ “control device” mean ~~recording equipment~~ “control device or its components”. No type approval is required for the cable(s) linking the motion sensor to the VU, the external GNSS facility to the VU or the external remote communication facility to the VU. The paper, for use by the ~~recording equipment~~ control device, shall be considered as a component of the ~~recording equipment~~ control device.

Any manufacturer may ask for type approval of its control device component(s) with any ~~type of motion sensor external GNSS facility and vice versa~~ other control device component(s), provided each component complies with the requirements of this Appendix. Alternately, manufacturers may also ask for type approval of ~~recording equipment~~ control devices.

~~Recording equipment~~ Vehicle units have variants in components assembly. Whatever the vehicle unit components assembly, the external antenna and (if applicable) the antenna splitter connected to the GNSS receiver or to the remote communication facility are not part of the vehicle unit type approval.

Nevertheless, manufacturers having obtained type approval for control device shall maintain a publicly available list of compatible antennas and splitters with each type approved vehicle unit, external GNSS facility and external remote communication facility.

(425) A control device shall be submitted for approval complete with any integrated additional devices.

(426) Type approval of ~~recording equipment~~ control devices and of tachograph cards shall include security related tests, functional tests and interoperability tests. Positive results to each of these tests are stated by an appropriate certificate.

(427) ~~Member States Contracting Parties~~ type approval authorities will not grant a type approval certificate as long as they do not hold:

- a security certificate (if requested by this sub-appendix),
- a functional certificate,

- and an interoperability certificate (**if requested by this sub-appendix**)

for the ~~recording equipment~~ **control device** or the tachograph card, subject of the request for type approval.

(428) Any modification in software or hardware of the equipment or in the nature of materials used for its manufacture shall, before being used, be notified to the authority which granted type-approval for the equipment. This authority shall confirm to the manufacturer the extension of the type approval, or may require an update or a confirmation of the relevant functional, security and/or interoperability certificates.

(429) Procedures to ~~update~~**update** in-situ ~~recording equipment~~ **control device** software shall be approved by the authority which granted type approval for the ~~recording equipment~~ **control device**. Software ~~update~~**update** must not alter nor delete any driver activity data stored in the ~~recording equipment~~ **control device**. Software may be ~~updated~~**updated** only under the responsibility of the equipment manufacturer.

(430) Type approval of software modifications aimed to ~~update~~**update** a previously type approved ~~recording equipment~~ **control device** may not be refused if such modifications only apply to functions not specified in this ~~Annex~~**Appendix**. Software ~~update~~**update** of a ~~recording equipment~~ **control device** may exclude the introduction of new character sets, if not technically feasible.

8.2. Security certificate

(431) The security certificate is delivered in accordance with the provisions of ~~Appendix~~ **Sub-appendix** 10 of this ~~Annex~~ ~~Recording equipment~~ ~~Appendix~~. **Control device** components to be certified are vehicle unit, motion sensor, external GNSS facility and tachograph cards.

(432) In the exceptional circumstance that the security certification authorities refuse to certify new equipment on the ground of obsolescence of the security mechanisms, type approval shall continue to be granted only in these specific and exceptional circumstances, and when no alternative solution, compliant with ~~Regulation~~ **this Agreement**, exists.

(433) In this circumstance the ~~Member State~~ ~~Contracting Party~~ concerned shall, without delay, inform the ~~European Commission~~, which shall ~~other Contracting Parties~~, **so that** within twelve calendar months of the grant of the type approval, a procedure to ensure that the level of security is restored to its original levels **is launched**.

8.3. Functional certificate

(434) Each candidate for type approval shall provide the ~~Member State's~~ **Contracting Party's** type approval authority with all the material and documentation that the authority deems necessary.

(435) Manufacturers shall provide the relevant samples of type approval candidate products and associated documentation required by laboratories appointed to perform functional tests, and within one month of the request being made. Any costs resulting from this request shall be borne by the requesting entity. Laboratories shall treat all commercially sensitive information in confidence.

(436) A functional certificate shall be delivered to the manufacturer only after all functional tests specified in ~~Appendix~~ **Sub-appendix** 9, at least, have been successfully passed.

(437) The type approval authority delivers the functional certificate. This certificate shall indicate, in addition to the name of its beneficiary and the identification of the model, a detailed list of the tests performed and the results obtained.

(438) The functional certificate of any ~~recording equipment~~ **control device** component shall also indicate the type approval numbers of the other type approved compatible ~~recording equipment~~ **control device** components tested for its certification.

(439) The functional certificate of any ~~recording equipment~~ **control device** component shall also indicate the ISO or CEN standard against which the functional interface has been certified.

8.4. Interoperability certificate

(440) Interoperability tests are carried out by a single ~~laboratory under the authority and responsibility of the European Commission~~ **competent body**.

(441) The laboratory shall register interoperability test requests introduced by manufacturers in the chronological order of their arrival.

(442) Requests will be officially registered only when the laboratory is in possession of:

- the entire set of material and documents necessary for such interoperability tests,
- the corresponding security certificate,
- the corresponding functional certificate,

The date of the registration of the request shall be notified to the manufacturer.

(443) No interoperability tests shall be carried out by the laboratory, for ~~recording equipment~~ **control devices** or tachograph cards that have not ~~been passed the vulnerability analysis of their granted~~ **been passed the vulnerability analysis of their granted**—a security ~~evaluation certificate~~ **evaluation certificate** and a functional ~~certificate evaluation~~ **certificate evaluation**, except in the exceptional circumstances described in ~~Requirement~~ **Requirement** 432.

(444) Any manufacturer requesting interoperability tests shall commit to leave to the laboratory in charge of these tests the entire set of material and documents which he provided to carry out the tests.

(445) The interoperability tests shall be carried out, in accordance with the provisions of ~~Appendix~~ **Sub-appendix** 9 of this ~~Annex~~ **Appendix**, with respectively all the types of ~~recording equipment~~ **control device** or tachograph cards:

- for which type approval is still valid or,
- for which type approval is pending and that have a valid interoperability certificate.

(446) The interoperability tests shall cover all generations of ~~recording equipment~~ **control device** or tachograph cards still in use.

(447) The interoperability certificate shall be ~~issued/delivered~~ by the laboratory to the manufacturer only after all required interoperability tests have been successfully passed **and after the manufacturer has shown that both a valid functional certificate and a valid security certificate for the product has been granted, except in the exceptional circumstances described in requirement 432.**

(448) If the interoperability tests are not successful with one or more of the ~~recording equipment~~ **control device** or tachograph card(s), the interoperability certificate shall not be delivered, until the requesting manufacturer has realised the necessary modifications and has succeeded the interoperability tests. The laboratory shall identify the cause of the problem

with the help of the manufacturers concerned by this interoperability fault and shall attempt to help the requesting manufacturer in finding a technical solution. In the case where the manufacturer has modified its product, it is the manufacturer's responsibility to ascertain from the relevant authorities that the security certificate and the functional certificates are still valid.

(449) The interoperability certificate is valid for six months. It is revoked at the end of this period if the manufacturer has not received a corresponding type approval certificate. It is forwarded by the manufacturer to the type approval authority of the ~~Member States~~ **Contracting Party** who has delivered the functional certificate.

(450) Any element that could be at the origin of an interoperability fault shall not be used for profit or to lead to a dominant position.

8.5. Type approval certificate

(451) The type approval authority of the ~~Member State~~ **Contracting Party** may deliver the type approval certificate as soon as it holds the three required certificates.

(452) The type approval certificate of any ~~recording equipment~~ **control device** component shall also indicate the type approval numbers of the other type approved interoperable ~~recording equipment~~ **control device**.

(453) The type approval certificate shall be copied by the type approval authority to the laboratory in charge of the interoperability tests at the time of deliverance to the manufacturer.

(454) The laboratory competent for interoperability tests shall run a public web site on which will be updated the list of ~~recording equipment~~ **control device** or tachograph cards models:

- for which a request for interoperability tests have been registered,
- having received an interoperability certificate (even provisional),
- having received a type approval certificate.

~~7.6 Exceptional procedure: first interoperability certificates for 2nd generation recording equipment and tachograph cards~~

~~455) Until four months after a first couple of 2nd generation recording equipment and 2nd generation tachograph cards (driver, workshop, control and company cards) have been certified to be interoperable, any interoperability certificate delivered (including the first ones), regarding requests registered during this period, shall be considered provisional.~~

~~456) If at the end of this period, all products concerned are mutually interoperable, all corresponding interoperability certificates shall become definitive.~~

~~457) If during this period, interoperability faults are found, the laboratory in charge of interoperability tests shall identify the causes of the problems with the help of all manufacturers involved and shall invite them to realize the necessary modifications.~~

~~458) If at the end of this period, interoperability problems still remain, the laboratory in charge of interoperability tests, with the collaboration of the manufacturers concerned and with the type approval authorities who delivered the corresponding functional certificates shall find out the causes of the interoperability faults and establish which modifications should be made by each of the manufacturers concerned. The search for technical solutions shall last for a maximum of two months, after which, if no common solution is found, the Commission, after having consulted the laboratory in charge of interoperability tests, shall~~

~~decide which equipment(s) and cards get a definitive interoperability certificate and state the reasons why.~~

~~459) — Any request for interoperability tests, registered by the laboratory between the end of the four month period after the first provisional interoperability certificate has been delivered and the date of the decision by the Commission referred to in requirement 455, shall be postponed until the initial interoperability problems have been solved. Those requests are then processed in the chronological order of their registration~~

~~ANNEX~~

APPENDIX

APPROVAL MARK AND CERTIFICATE

I. APPROVAL MARK

1. The approval mark shall be made up of:

(a) a rectangle, within which shall be placed the letter 'e' followed by a distinguishing number or letter for the country which has issued the approval in accordance with the following conventional signs:

Drafting note: All Contracting Parties to be listed

Belgium	6,
Bulgaria	34,
Czech Republic	8,
Denmark	18,
Germany	1,
Estonia	29,
Ireland	24,
Greece	23,
Spain	9,
France	2,
Croatia	25,
Italy	3,
Cyprus	CY,
Latvia	32,
Lithuania	36,
Luxembourg	13,
Hungary	7,
Malta	MT,
Netherlands	4,
Austria	12,
Poland	20,
Portugal	21,
Romania	19,

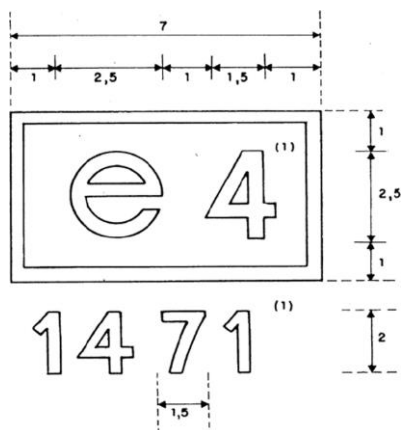
Slovenia	26,
Slovakia	27,
Finland	17,
Sweden	5,
United Kingdom	11,

(a) a rectangle, within which shall be placed the letter 'e' followed by a distinguishing number or letter for the country which has issued the approval in accordance with the following conventional signs:

(b) an approval number corresponding to the number of the approval certificate drawn up for the prototype of the ~~recording equipment~~ **control device** or ~~to the number of a record sheet or the tachograph card~~, placed at any point within the immediate proximity of that rectangle.

2. The approval mark shall be shown on the descriptive plaque of each set of equipment and on each record sheet and on each tachograph card. It must be indelible and must always remain clearly legible.

3. The dimensions of the approval mark drawn below ⁽¹⁾ are expressed in millimetres, these dimensions being minima. The ratios between the dimensions must be maintained.



⁽¹⁾ These figures are shown for guidance only.

II. APPROVAL CERTIFICATE FOR ANALOGUE TACHOGRAPHS

A ~~Member State~~ **Contracting Party** which has granted approval shall issue the applicant with an approval certificate, the model of which is given below. When informing other ~~Member States~~ **Contracting Parties** of approvals issued or, if the occasion should arise, withdrawn, a ~~Member State~~ **Contracting Party** shall use copies of that certificate.

APPROVAL CERTIFICATE

- Name of competent administration
- Notification concerning ⁽¹⁾:
- approval of a type of ~~recording equipment~~ **control device**
- withdrawal of approval of a type of ~~recording equipment~~ **control device**
- approval of a model record sheet
- withdrawal of approval of a model record sheet

Approval No:

1. Trade mark or name
2. Name of type or model
3. Name of manufacturer
.....
4. Address of manufacturer
5. Submitted for approval on
6. Tested at
7. Date and number of the test(s)
8. Date of approval
9. Date of withdrawal of approval
10. Type or types of ~~recording equipment~~ **control device** in which sheet is designed to be used
11. Place
12. Date
13. Descriptive documents ~~annexed~~ **appended**
14. Remarks (including the position of seals if applicable)

(Signature)

(1) Delete items not applicable

III. APPROVAL CERTIFICATE FOR DIGITAL TACHOGRAPHS

A ~~Member State Contracting Party~~ which has granted approval shall issue the applicant with an approval certificate, the model of which is given below. When informing other ~~Member States Contracting Parties~~ of approvals issued or, if the occasion should arise, withdrawn, a ~~Member State Contracting Party~~ shall use copies of that certificate.

APPROVAL CERTIFICATE FOR DIGITAL TACHOGRAPHS

Name of competent administration

Notification concerning ⁽¹⁾:

- | | |
|---|--|
| <input type="checkbox"/> approval of:

component ⁽²⁾ | <input type="checkbox"/> withdrawal of approval of:

<input type="checkbox"/> recording equipment control device model
<input type="checkbox"/> _____ recording equipment control device

<input type="checkbox"/> a driver's card
<input type="checkbox"/> a workshop card
<input type="checkbox"/> a company card
<input type="checkbox"/> a controller's card |
|---|--|

Approval No:

1. Manufacturing brand or trademark
2. Name of model
3. Name of manufacturer
4. Address of manufacturer
5. Submitted for approval ~~for on~~
6. Laboratory(-ies)
7. Date and number of test report
8. Date of approval
9. Date of withdrawal of approval
10. Model of ~~recording equipment control device~~(s) with which the component is designed to be used
11. Place
12. Date
13. Descriptive documents ~~annexed~~ **appendixed**
14. Remarks (including the position of seals if applicable)

(Signature)

- (1) Tick the relevant boxes.
- (2) Specify the component dealt with in the notification.

IV. APPROVAL CERTIFICATE FOR SMART TACHOGRAPHS

A ~~Member State~~ **Contracting Party** which has granted approval shall issue the applicant with an approval certificate, the model of which is given below. When informing other ~~Member States~~ **Contracting Parties** of approvals issued or, if the occasion should arise, withdrawn, a ~~Member State~~ **Contracting Party** shall use copies of that certificate.

APPROVAL CERTIFICATE FOR SMART TACHOGRAPHS

Name of competent administration

Notification concerning ⁽¹⁾:

- approval of: withdrawal of approval of:
- ~~recording equipment control device~~ model
 - ~~recording equipment control device~~ component ⁽²⁾
 - a driver's card
 - a workshop card
 - a company card
 - a controller's card

Approval No:

1. Manufacturing brand or trademark
2. Name of model
3. Name of manufacturer
4. Address of manufacturer
5. Submitted for approval ~~for on~~
6. a Test laboratory for functional certification.....
- b Test laboratory for security certification
- c Test laboratory for interoperability certification.....
7. a Date and number of functional certificate
- b Date and number of security certificate.....
- c Date and number of interoperability certificate
8. Date of approval
9. Date of withdrawal of approval
10. Model of ~~recording equipment control device~~(s) with which the component is designed to be used
11. Place
12. Date
13. Descriptive documents ~~annexed~~ **appendixed**
14. Remarks (including the position of seals if applicable)

(Signature)

- (1) Tick the relevant boxes.
- (2) Specify the component dealt with in the notification.

Appendix Sub-appendix 1. Data dictionary

TABLE OF CONTENT

1. Introduction.....	113
1.1. Approach for definitions of data types	113
1.2. References.....	113
2. Data Type Definitions	113
2.1. ActivityChangeInfo.....	114
2.2. Address	116
2.3. AESKey	116
2.4. AES128Key	116
2.5. AES192Key	116
2.6. AES256Key	117
2.7. BCDString.....	117
2.8. CalibrationPurpose	117
2.9. CardActivityDailyRecord	118
2.10. CardActivityLengthRange	118
2.11. CardApprovalNumber	118
2.11a. CardBorderCrossing	119
2.11b. CardBorderCrossingRecord	119
2.12. CardCertificate	119
2.13. CardChiplIdentification	119
2.14. CardConsecutiveIndex.....	120
2.15. CardControlActivityDataRecord.....	120
2.16. CardCurrentUse.....	120
2.17. CardDriverActivity.....	120
2.18. CardDrivingLicenceInformation	121
2.19. CardEventData.....	121
2.20. CardEventRecord.....	122
2.21. CardFaultData.....	123
2.22. CardFaultRecord	123
2.23. CardIccIdentification	124
2.24. CardIdentification	124
2.24a. CardLoadTypeEntries	125
2.24b. CardLoadTypeEntryRecord.....	125
2.24c. CardLoadUnloadOperations	125
2.24d. CardLoadUnloadRecord	126
2.25. CardMACertificate	127
2.26. CardNumber.....	127
2.26a. CardPlaceAuthDailyWorkPeriod	127
2.27. CardPlaceDailyWorkPeriod.....	128

2.28.	CardPrivateKey.....	128
2.29.	CardPublicKey	128
2.30.	CardRenewalIndex	128
2.31.	CardReplacementIndex.....	129
2.32.	CardSignCertificate	129
2.33.	CardSlotNumber.....	129
2.34.	CardSlotsStatus	129
2.35.	CardSlotsStatusRecordArray	129
2.36.	CardStructureVersion	130
2.37.	CardVehicleRecord	130
2.38.	CardVehiclesUsed.....	131
2.39.	CardVehicleUnitRecord	132
2.40.	CardVehicleUnitsUsed.....	132
2.41.	Certificate.....	133
2.42.	CertificateContent.....	133
2.43.	CertificateHolderAuthorisation	133
2.44.	CertificateRequestID	134
2.45.	CertificationAuthorityKID.....	135
2.46.	CompanyActivityData	135
2.47.	CompanyActivityType	136
2.48.	CompanyCardApplicationIdentification	136
2.48a.	CompanyCardApplicationIdentificationV2.....	136
2.49.	CompanyCardHolderIdentification	136
2.50.	ControlCardApplicationIdentification	137
2.50a.	ControlCardApplicationIdentificationV2	137
2.51.	ControlCardControlActivityData	137
2.52.	ControlCardHolderIdentification	138
2.53.	ControlType.....	139
2.54.	CurrentDateTime	139
2.55.	CurrentDateTimeRecordArray	140
2.56.	DailyPresenceCounter	141
2.57.	Datef	141
2.58.	DateOfDayDownloaded	141
2.59.	DateOfDayDownloadedRecordArray	141
2.60.	Distance	141
2.60a.	DownloadInterfaceVersion	142
2.61.	DriverCardApplicationIdentification	142
2.61a.	DriverCardApplicationIdentificationV2	143
2.62.	DriverCardHolderIdentification	144
2.63.	DSRC Security Data Reserved for future use	144
2.64.	EGFCertificate	Error! Bookmark not defined. 145

2.65.	<i>EmbedderIcAssemblerId</i>	145
2.66.	<i>EntryTypeDailyWorkPeriod</i>	146
2.67.	<i>EquipmentType</i>	146
2.68.	<i>EuropeanPublicKey</i>	147
2.69.	<i>EventFaultRecordPurpose</i>	148
2.70.	<i>EventFaultType</i>	148
2.71.	<i>ExtendedSealIdentifier</i>	153
2.72.	<i>ExtendedSerialNumber</i>	154
2.73.	<i>FullCardNumber</i>	155
2.74.	<i>FullCardNumberAndGeneration</i>	155
2.75.	<i>Generation</i>	155
2.76.	<i>GeoCoordinates</i>	155
2.77.	<i>GNSSAccuracy</i>	156
2.78.	<i>GNSSCountinousDriving</i> GNSSAccumulatedDriving	156
2.79.	<i>GNSSCountinousDrivingRecord</i> GNSSAccumulatedDrivingRecord	156
2.79a.	<i>GNSSAuthAccumulatedDriving</i>	157
2.79b.	<i>GNSSAuthStatusADRecord</i>	157
2.79c.	<i>GNSSPlaceAuthRecord</i>	157
2.80.	<i>GNSSPlaceRecord</i>	158
2.81.	<i>HighResOdometer</i>	158
2.82.	<i>HighResTripDistance</i>	159
2.83.	<i>HolderName</i>	159
2.84.	<i>InternalGNSSReceiver</i> Reserved for future use	159
2.85.	<i>K-ConstantOfRecordingEquipment</i>	159
2.86.	<i>KeyIdentifier</i>	160
2.87.	<i>KMWCKey</i>	160
2.88.	<i>Language</i>	160
2.89.	<i>LastCardDownload</i>	160
2.89a.	<i>LengthOfFollowingData</i>	161
2.90.	<i>LinkCertificate</i>	161
2.90a.	<i>LoadType</i>	161
2.91.	<i>L-TyreCircumference</i>	161
2.92.	<i>MAC</i>	162
2.93.	<i>ManuallInputFlag</i>	162
2.94.	<i>ManufacturerCode</i>	162
2.95.	<i>ManufacturerSpecificEventFaultData</i>	162
2.96.	<i>MemberStateCertificate</i>	162
2.97.	<i>MemberStateCertificateRecordArray</i>	162
2.98.	<i>MemberStatePublicKey</i>	163
2.99.	<i>Name</i>	163
2.100.	<i>NationAlpha</i>	163

2.101. NationNumeric.....	164
2.101a. NoOfBorderCrossingRecords.....	164
2.102. NoOfCalibrationRecords	164
2.103. NoOfCalibrationsSinceDownload.....	164
2.104. NoOfCardPlaceRecords	165
2.105. NoOfCardVehicleRecords	165
2.106. NoOfCardVehicleUnitRecords	165
2.107. NoOfCompanyActivityRecords	165
2.108. NoOfControlActivityRecords	165
2.109. NoOfEventsPerType	165
2.110. NoOfFaultsPerType	165
2.111. NoOfGNSSCDRecords NoOfGNSSADRecords	166
2.111a. NoOfLoadUnloadRecords.....	166
2.112. NoOfSpecificConditionRecords.....	166
2.112a. NoOfLoadTypeEntryRecords	166
2.113. OdometerShort	166
2.114. OdometerValueMidnight	166
2.114a. OperationType	166
2.115. OdometerValueMidnightRecordArray	167
2.116. OverspeedNumber	167
2.116a. PlaceAuthRecord.....	167
2.116b. PlaceAuthStatusRecord.....	168
2.117. PlaceRecord	168
2.117a. PositionAuthenticationStatus	169
2.118. PreviousVehicleInfo.....	169
2.119. PublicKey.....	170
2.120. RecordType	171
2.121. RegionAlpha.....	172
2.122. RegionNumeric.....	172
2.123. RemoteCommunicationModuleSerialNumber	173
2.124. RSAKeyModulus	173
2.125. RSAKeyPrivateExponent.....	173
2.126. RSAKeyPublicExponent.....	174
2.127. RtmData.....	174
2.128. SealDataCard	174
2.129. SealDataVu	174
2.130. SealRecord	174
2.131. SensorApprovalNumber	175
2.132. SensorExternalGNSSApprovalNumber	175
2.133. SensorExternalGNSSCoupledRecord	175
2.134. SensorExternalGNSSIdentification	176

2.135.	<i>SensorExternalGNSSInstallation</i>	176
2.136.	<i>SensorExternalGNSSOSIdentifier</i>	177
2.137.	<i>SensorExternalGNSSSCIdentifier</i>	177
2.138.	<i>SensorGNSSCouplingDate</i>	177
2.139.	<i>SensorGNSSSerialNumber</i>	177
2.140.	<i>SensorIdentification</i>	177
2.141.	<i>SensorInstallation</i>	178
2.142.	<i>SensorInstallationSecData</i>	178
2.143.	<i>SensorOSIdentifier</i>	179
2.144.	<i>SensorPaired</i>	179
2.145.	<i>SensorPairedRecord</i>	179
2.146.	<i>SensorPairingDate</i>	180
2.147.	<i>SensorSCIdentifier</i>	180
2.148.	<i>SensorSerialNumber</i>	180
2.149.	<i>Signature</i>	180
2.150.	<i>SignatureRecordArray</i>	180
2.151.	<i>SimilarEventsNumber</i>	181
2.152.	<i>SpecificConditionRecord</i>	181
2.153.	<i>SpecificConditions</i>	181
2.154.	<i>SpecificConditionType</i>	182
2.155.	<i>Speed</i>	182
2.156.	<i>SpeedAuthorised</i>	182
2.157.	<i>SpeedAverage</i>	183
2.158.	<i>SpeedMax</i>	183
2.158a.	<i>TachographCardsGen1Suppression</i>	183
2.159.	<i>TachographPayload</i>	183
2.160.	<i>Tachographpayloadencrypted</i> Reserved for future use	183
2.161.	<i>TDesSessionKey</i>	184
2.162.	<i>TimeReal</i>	184
2.163.	<i>TyreSize</i>	184
2.164.	<i>VehicleIdentificationNumber</i>	184
2.165.	<i>VehicleIdentificationNumberRecordArray</i>	184
2.166.	<i>VehicleRegistrationIdentification</i>	185
2.166a.	<i>VehicleRegistrationIdentificationRecordArray</i>	185
2.167.	<i>VehicleRegistrationNumber</i>	185
2.168.	<i>VehicleRegistrationNumberRecordArray</i>	186
2.169.	<i>VuAbility</i>	186
2.170.	<i>VuActivityDailyData</i>	187
2.171.	<i>VuActivityDailyRecordArray</i>	187
2.172.	<i>VuApprovalNumber</i>	188
2.173.	<i>VuCalibrationData</i>	188

2.174.	<i>VuCalibrationRecord</i>	188
2.175.	<i>VuCalibrationRecordArray</i>	190
2.176.	<i>VuCardIWData</i>	191
2.177.	<i>VuCardIWRecord</i>	192
2.178.	<i>VuCardIWRecordArray</i>	193
2.179.	<i>VuCardRecord</i>	194
2.180.	<i>VuCardRecordArray</i>	194
2.181.	<i>VuCertificate</i>	195
2.182.	<i>VuCertificateRecordArray</i>	195
2.183.	<i>VuCompanyLocksData</i>	195
2.184.	<i>VuCompanyLocksRecord</i>	195
2.185.	<i>VuCompanyLocksRecordArray</i>	196
2.185a.	<i>VuConfigurationLengthRange</i>	196
2.186.	<i>VuControlActivityData</i>	197
2.187.	<i>VuControlActivityRecord</i>	197
2.188.	<i>VuControlActivityRecordArray</i>	198
2.189.	<i>VuDataBlockCounter</i>	198
2.190.	<i>VuDetailedSpeedBlock</i>	199
2.191.	<i>VuDetailedSpeedBlockRecordArray</i>	199
2.192.	<i>VuDetailedSpeedData</i>	199
2.192a.	<i>VuDigitalMapVersion</i>	199
2.193.	<i>VuDownloadablePeriod</i>	200
2.194.	<i>VuDownloadablePeriodRecordArray</i>	200
2.195.	<i>VuDownloadActivityData</i>	200
2.196.	<i>VuDownloadActivityDataRecordArray</i>	201
2.197.	<i>VuEventData</i>	201
2.198.	<i>VuEventRecord</i>	202
2.199.	<i>VuEventRecordArray</i>	203
2.200.	<i>VuFaultData</i>	204
2.201.	<i>VuFaultRecord</i>	204
2.202.	<i>VuFaultRecordArray</i>	205
2.203.	<i>VuGNSSDRECORDVuGNSSADRecord</i>	206
2.203a.	<i>VuBorderCrossingRecord</i>	207
2.203b.	<i>VuBorderCrossingRecordArray</i>	207
2.204.	<i>VuGNSSDRECORDArray VuGNSSADRecordArray</i>	208
2.204a.	<i>VuGnssMaximalTimeDifference</i>	208
2.205.	<i>VuIdentification</i>	209
2.206.	<i>VuIdentificationRecordArray</i>	210
2.207.	<i>VuITSConsentRecord</i>	210
2.208.	<i>VuITSConsentRecordArray</i>	210
2.208a.	<i>VuLoadUnloadRecord</i>	

2.208b. VuLoadUnloadRecordArray	
2.209. VuManufacturerAddress	211
2.210. VuManufacturerName	212
2.211. VuManufacturingDate	212
2.212. VuOverSpeedingControlData	212
2.213. VuOverSpeedingControlDataRecordArray	212
2.214. VuOverSpeedingEventData	214
2.215. VuOverSpeedingEventRecord	214
2.216. VuOverSpeedingEventRecordArray	215
2.217. VuPartNumber	215
2.218. VuPlaceDailyWorkPeriodData	215
2.219. VuPlaceDailyWorkPeriodRecord	216
2.220. VuPlaceDailyWorkPeriodRecordArray	217
2.221. VuPrivateKey	217
2.222. VuPublicKey	217
2.222a. VuRtcTime	217
2.223. VuSerialNumber	218
2.224. VuSoftInstallationDate	218
2.225. VuSoftwareIdentification	218
2.226. VuSoftwareVersion	218
2.227. VuSpecificConditionData	218
2.228. VuSpecificConditionRecordArray	218
2.229. VuTimeAdjustmentData	219
2.230. VutimeadjustmentGnssrecordarray Reserved for future use	219
2.231. Reserved for future use	219
2.232. VuTimeAdjustmentRecord	219
2.233. VuTimeAdjustmentRecordArray	221
2.234. WorkshopCardApplicationIdentification	222
2.234a. WorkshopCardApplicationIdentificationV2	223
2.234b. WorkshopCardCalibrationAddData	223
2.234c. WorkshopCardCalibrationAddDataRecord	224
2.235. WorkshopCardCalibrationData	224
2.236. WorkshopCardCalibrationRecord	224
2.237. WorkshopCardHolderIdentification	226
2.238. WorkshopCardPIN	227
2.239. W-VehicleCharacteristicConstant	227
2.240. VuPowerSupplyInterruptionRecord	227
2.241. VuPowerSupplyInterruptionRecordArray	228
2.242. VuSensorExternalGNSSCoupledRecordArray	229
2.243. VuSensorPairedRecordArray	229
3. Value and size range definitions	229

4. *Character sets*..... 229

5. *Encoding*..... 230

6. *Object Identifiers und Application Identifiers*..... 230

6.1. *Object Identifiers*..... 230

6.2. *Application Identifiers*..... 231

1. Introduction

This **sub**-appendix specifies data formats, data elements, and data structures for use within the ~~recording equipment~~ **control device** and tachograph cards.

1.1 Approach for definitions of data types

This **sub**-appendix uses Abstract Syntax Notation One (ASN.1) to define data types. This enables simple and structured data to be defined without implying any specific transfer syntax (encoding rules) which will be application and environment dependent.

ASN.1 type naming conventions are done in accordance with ISO/IEC 8824-1. This implies that:

- where possible, the meaning of the data type is implied through the names being selected,
- where a data type is a composition of other data types, the data type name is still a single sequence of alphabetical characters commencing with a capital letter, however capitals are used within the name to impart the corresponding meaning,
- in general, the data types names are related to the name of the data types from which they are constructed, the equipment in which data is stored and the function related to the data.

If an ASN.1 type is already defined as part of another standard and if it is relevant for usage in the ~~recording equipment~~ **control device**, then this ASN.1 type will be defined in this sub-appendix.

To enable several types of encoding rules, some ASN.1 types in this **sub**-appendix are constrained by value range identifiers. The value range identifiers are defined in paragraph 3 and ~~Appendix~~ **Sub-appendix 2**.

1.2. References

The following references are used in this ~~Appendix~~ **sub-appendix**:

ISO 639	Code for the representation of names of languages. First Edition: 1988.
ISO 3166	Codes for the representation of names of countries and their subdivisions – Part 1: Country codes, 2013
ISO 3779	Road vehicles - Vehicle identification number (VIN) - Content and structure. 2009
ISO/IEC 7816-5	Identification cards - Integrated circuit cards - Part 5: Registration of application providers .Second edition: 2004.
ISO/IEC 7816-6	Identification cards - Integrated circuit cards – Part 6: Interindustry data elements for interchange, 2004 + Technical Corrigendum 1: 2006
ISO/IEC 8824-1	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation. 2008 + Technical Corrigendum 1: 2012 and Technical Corrigendum 2: 2014.
ISO/IEC 8825-2	Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER). 2008.

- ISO/IEC 8859-1 Information technology - 8 bit single-byte coded graphic character sets
- Part 1: Latin alphabet No.1. First edition: 1998.
- ISO/IEC 8859-7 Information technology - 8 bit single-byte coded graphic character sets
- Part 7: Latin/Greek alphabet. 2003.
- ISO 16844-3 Road vehicles - Tachograph systems - Motion Sensor Interface. 2004 +
Technical Corrigendum 1: 2006.
- TR-03110-3 BSI / ANSSI Technical Guideline TR-03110-3, Advanced Security
Mechanisms for Machine Readable Travel Documents and eIDAS Token - Part 3 Common
Specifications, version 2.20, 3. February 2015

2. ~~Data Type Definitions~~ type definitions

For any of the following data types, the default value for an “unknown” or a “not applicable” content will consist in filling the data element with Hex ‘FF’ bytes, **unless otherwise specified**.

All data types are used for Generation 1 and Generation 2 applications unless otherwise specified.

For card Data types only used for Generation 2, version 2 applications are indicated. data types used for Generation 1 and Generation 2 applications, the size specified in this sub-appendix is the one for Generation 2 application. The size for Generation 1 application is supposed to be already known by the reader. The Appendix 1C requirement numbers related to such data types cover both Generation 1 and Generation 2 applications.

Card data types not defined for Generation 1 cards are not stored in Generation 1 application of Generation 2 cards. In particular:

- **Type approval numbers stored in Generation 1 application of Generation 2 cards are truncated to the 8 first characters where needed,**
- **Only the ‘FERRY / TRAIN CROSSING begin’ of a ‘FERRY / TRAIN CROSSING’ specific condition is stored in Generation 1 application of Generation 2 cards.**

2.1. ActivityChangeInfo

This data type enables to code, within a two bytes word, a slot status at 00:00 and/or a driver status at 00:00 and/or changes of activity and/or changes of driving status and/or changes of card status for a driver or a co-driver. This data type is related to ~~Annex~~ **Appendix 1C** requirements 105, 266, 291, 320, 321, 343, and 344.

ActivityChangeInfo ::= OCTET STRING (SIZE(2))

Value assignment – Octet Aligned: ‘sepaatttttttt’B (16 bits)

For Data Memory recordings (or slot status):

's'B	Slot: '0'B: DRIVER, '1'B: CO-DRIVER,
'c'B	Driving status: '0'B: SINGLE, '1'B: CREW,
'p'B	Driver (or workshop) card status in the relevant slot: '0'B: INSERTED, a card is inserted, '1'B: NOT INSERTED, no card is inserted (or a card is withdrawn),
'aa'B	Activity: '00'B: BREAK/REST, '01'B: AVAILABILITY, '10'B: WORK, '11'B: DRIVING,
'tttttttt'B	Time of the change: Number of minutes since 00h00 on the given day.

For Driver (or Workshop) card recordings (and driver status):

's'B	Slot (not relevant when 'p'=1 except note below): '0'B: DRIVER, '1'B: CO-DRIVER,
'c'B	Driving status (case 'p'=0) or Following activity status (case 'p'=1): '0'B: SINGLE, '0'B: UNKNOWN '1'B: CREW, '1'B: KNOWN (=manually entered)
'p'B	Card status: '0'B: INSERTED, the card is inserted in a recording equipment control device , '1'B: NOT INSERTED, the card is not inserted (or the card is withdrawn),
'aa'B	Activity (not relevant when 'p'=1 and 'c'=0 except note below): '00'B: BREAK/REST, '01'B: AVAILABILITY, '10'B: WORK, '11'B: DRIVING,
'tttttttt'B	Time of the change: Number of minutes since 00h00 on the given day.

Note for the case 'card withdrawal':

When the card is withdrawn:

- 's' is relevant and indicates the slot from which the card is withdrawn,
- 'c' must be set to 0,
- 'p' must be set to 1,
- 'aa' must code the current activity selected at that time,

As a result of a manual entry, the bits 'c' and 'aa' of the word (stored in a card) may be overwritten later to reflect the entry.

2.2. Address

An address.

```
Address ::= SEQUENCE {  
codePage    INTEGER (0..255),  
address     OCTET STRING (SIZE(35))  
}
```

codePage specifies a character set defined in Chapter 4,

address is an address encoded using the specified character set.

2.3. AESKey

Generation 2:

An AES key with a length of 128, 192 or 256 bits.

```
AESKey ::= CHOICE {  
aes128Key  AES128Key,  
aes192Key  AES192Key,  
aes256Key  AES256Key  
}
```

Value assignment: not further specified.

2.4. AES128Key

Generation 2:

An AES128 key.

```
AES128Key ::= SEQUENCE {  
length INTEGER(0..255),  
aes128Key OCTET STRING (SIZE(16))  
}
```

length denotes the length of the AES128 key in octets. **aes128Key** is an AES key with a length of 128 bits.

Value assignment:

The length shall have the value 16.

2.5. AES192Key

Generation 2:

An AES192 key.

```
AES192Key ::= SEQUENCE {  
length INTEGER(0..255),
```

```

aes192Key  OCTET STRING (SIZE(24))
          }

```

length denotes the length of the AES192 key in octets.

aes192Key is an AES key with a length of 192 bits.

Value assignment:

The length shall have the value 24.

2.6 AES256Key

Generation 2:

An AES256 key.

```

AES256Key ::= SEQUENCE {
length INTEGER(0..255),
aes256Key  OCTET STRING (SIZE(32))
          }

```

length denotes the length of the AES256 key in octets.

aes256Key is an AES key with a length of 256 bits.

Value assignment:

The length shall have the value 32.

2.7 BCDString

BCDString is applied for Binary Code Decimal (BCD) representation. This data type is used to represent one decimal digit in one semi octet (4 bits). BCDString is based on the ISO/IEC 8824-1 'CharacterStringType'.

```

BCDString ::= CHARACTER STRING (WITH COMPONENTS {
identification ( WITH COMPONENTS {
fixed PRESENT } ) )

```

BCDString uses an "hstring" notation. The leftmost hexadecimal digit shall be the most significant semi octet of the first octet. To produce a multiple of octets, zero trailing semi octets shall be inserted, as needed, from the leftmost semi octet position in the first octet.

Permitted digits are : 0, 1, .. 9.

2.8 CalibrationPurpose

Code explaining why a set of calibration parameters was recorded. This data type is related to ~~Annex~~ **Appendix 1B** requirements 097 and 098 and ~~Annex Appendix 1C requirements~~ **requirement 119**.

```

CalibrationPurpose ::= OCTET STRING (SIZE(1))

```

Value assignment:

Generation 1:

- '00'H reserved value,
- '01'H activation: recording of calibration parameters known, at the moment of the VU activation,
- '02'H first installation: first calibration of the VU after its activation,
- '03'H installation: first calibration of the VU in the current vehicle,
- '04'H periodic inspection.

Generation 2:

In addition to generation 1 the following values are used:

- '05'H entry of VRN by company,
- '06'H time adjustment without calibration,
- '07'H to '7F'H RFU, '80'H to 'FF'H Manufacturer specific.

2.9. CardActivityDailyRecord

Information, stored in a card, related to the driver activities for a particular calendar day. This data type is related to ~~Annex~~ **Appendix 1C** requirements 266, 291, 320 and 343.

```

CardActivityDailyRecord ::      = SEQUENCE {
activityPreviousRecordLength    INTEGER(0..CardActivityLengthRange),
activityRecordLength            INTEGER(0..CardActivityLengthRange),
activityRecordDate              TimeReal,
activityDailyPresenceCounter    DailyPresenceCounter,
activityDayDistance             Distance,
activityChangeInfo              SET SIZE(1..1440) OF ActivityChangeInfo
}

```

activityPreviousRecordLength is the total length in bytes of the previous daily record. The maximum value is given by the length of the OCTET STRING containing these records (see CardActivityLengthRange ~~Appendix~~ **Sub-appendix 2** paragraph 4). When this record is the oldest daily record, the value of activityPreviousRecordLength must be set to 0.

activityRecordLength is the total length in bytes of this record. The maximum value is given by the length of the OCTET STRING containing these records.

activityRecordDate is the date of the record.

activityDailyPresenceCounter is the daily presence counter for the card this day.

activityDayDistance is the total distance travelled this day.

activityChangeInfo is the set of ActivityChangeInfo data for the driver this day. It may contain at maximum 1440 values (one activity change per minute). This set always includes the activityChangeInfo coding the driver status at 00:00.

2.10. CardActivityLengthRange

Number of bytes in a driver or a workshop card, available to store driver activity records.

CardActivityLengthRange ::= INTEGER(0..2¹⁶-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.11. CardApprovalNumber

Type approval number of the card.

CardApprovalNumber ::= IA5String(SIZE(8))

Value assignment:

The approval number shall be provided as published on the corresponding ~~European Commission web site, i.e. website run by the laboratory competent for interoperability tests, i.e.~~ for example including hyphens if any. The approval number shall be left-aligned.

2.11a. CardBorderCrossings

Generation 2, version 2:

Information, stored in a driver or workshop card, related to the border crossings of the vehicle when the latter has crossed the border of a country (Appendix 1C requirements 306f and 356f).

```
CardBorderCrossings ::= SEQUENCE {
    borderCrossingPointerNewestRecord          INTEGER
                                              (0..NoOfBorderCrossingRecords -1),
    cardBorderCrossingRecords                 SET
                                              (NoOfBorderCrossingRecords)
                                              OF CardBorderCrossingRecord
}
```

borderCrossingPointerNewestRecord is the index of the last updated card border crossing record.

Value assignment is the number corresponding to the numerator of the card border crossing record, beginning with '0' for the first occurrence of the card border crossing record in the structure.

cardBorderCrossingRecords is the set of card border crossing records.

2.11b. CardBorderCrossingRecordd

Generation 2, version 2:

Information, stored in a driver or workshop card, related to the border crossings of the vehicle when the latter has crossed the border of a country (Appendix 1C requirements 147b, 306e and 356e).

```
CardBorderCrossingRecord ::= SEQUENCE {
    countryLeft                NationNumeric,
    countryEntered              NationNumeric,
    gnssPlaceAuthRecord        GNSSPlaceAuthRecord,
    vehicleOdometerValue       OdometerShort
}
```

countryLeft is the country which was left by the vehicle, or 'no information available' according to Appendix 1C requirement 147b. 'Rest of the World' (NationNumeric code 'FF'H) shall be used when the vehicle unit is not able to determine the country where the vehicle is located (e.g. the current country is not part of the stored digital maps).

countryEntered is the country into which the vehicle has entered, or the country in which the vehicle is located at card insertion time. ‘Rest of the World’ (NationNumeric code ‘FF’H) shall be used when the vehicle unit is not able to determine the country where the vehicle is located (e.g. the current country is not part of the stored digital maps).

gnssPlaceAuthRecord contains information related to the position of the vehicle, when the vehicle unit has detected that the vehicle has crossed the border of a country, or ‘no information available’ according to requirement 147b of Appendix 1C, and its authentication status.

vehicleOdometerValue is the odometer value when the vehicle unit has detected that the vehicle has crossed the border of a country, or ‘no information available’ according to requirement 147b of Appendix 1C.

2.12. CardCertificate

Generation 1:
Certificate of the public key of a card.
CardCertificate ::= Certificate

2.13. CardChipIdentification

Information, stored in a card, related to the identification of the card’s Integrated Circuit (IC) (~~Annex~~ Appendix 1C requirement 249). The icSerialNumber together with the icManufacturingReferences identifies the card chip uniquely. The icSerialNumber alone does not uniquely identify the card chip.

```
CardChipIdentification ::= SEQUENCE {
    icSerialNumber          OCTET STRING (SIZE(4)),
    icManufacturingReferences OCTET STRING (SIZE(4))
}
```

icSerialNumber is the IC serial number.

icManufacturingReferences is the IC manufacturer specific identifier.

2.14. CardConsecutiveIndex

A card consecutive index (definition h)).

CardConsecutiveIndex ::= IA5String(SIZE(1))

Value assignment: (see ~~Annex~~ Appendix 1C chapter 7)

Order for increase: ‘0 , ..., 9, A , ..., Z, a , ..., z’

2.15. CardControlActivityDataRecord

Information, stored in a driver or workshop card, related to the last control the driver has been subject to (~~Annex~~ Appendix 1C requirements 274, 299, 327, and 350).

```
CardControlActivityDataRecord ::= SEQUENCE {
    controlType          ControlType,
```



```

controlTime                TimeReal,
controlCardNumber          FullCardNumber,
controlVehicleRegistration VehicleRegistrationIdentification,
controlDownloadPeriodBegin TimeReal,
controlDownloadPeriodEnd   TimeReal   }

```

controlType is the type of the control.

controlTime is the date and time of the control.

controlCardNumber is the FullCardNumber of the control officer having performed the control.

controlVehicleRegistration is the VRN and registering ~~Member State~~ **Contracting Party** of the vehicle in which the control happened.

controlDownloadPeriodBegin and **controlDownloadPeriodEnd** is the period downloaded, in case of downloading.

2.16. CardCurrentUse

Information about the actual usage of the card (~~Annex~~ **Appendix** 1C requirement 273, 298, 326, and 349).

```

CardCurrentUse ::= SEQUENCE {
sessionOpenTime      TimeReal,
sessionOpenVehicle   VehicleRegistrationIdentification
}

```

sessionOpenTime is the time when the card is inserted for the current usage. This element is set to zero at card removal.

sessionOpenVehicle is the identification of the currently used vehicle, set at card insertion. This element is set to zero at card removal.

2.17. CardDriverActivity

Information, stored in a driver or a workshop card, related to the activities of the driver (~~Annex~~ **Appendix** 1C requirements 267, 268, 292, 293, 321 and 344).

```

CardDriverActivity ::= SEQUENCE {
activityPointerOldestDayRecord    INTEGER(0.. CardActivityLengthRange-
1),
activityPointerNewestRecord      INTEGER(0.. CardActivityLengthRange-
1),
activityDailyRecords             OCTET STRING
                                 (SIZE(CardActivityLengthRange))
}

```

activityPointerOldestDayRecord is the specification of the begin of the storage place (number of bytes from the beginning of the string) of the oldest complete day record in the activityDailyRecords string. The maximum value is given by the length of the string.

activityPointerNewestRecord is the specification of the begin of the storage place (number of bytes from the beginning of the string) of the most recent day record in the activityDailyRecords string. The maximum value is given by the length of the string.

activityDailyRecords is the space available to store the driver activity data (data structure: CardActivityDailyRecord) for each calendar day where the card has been used.

Value assignment: this octet string is cyclically filled with records of CardActivityDailyRecord. At the first use storing is started at the first byte of the string. All new records are appended at the end of the previous one. When the string is full, storing continues at the first byte of the string independently of a break being inside a data element. Before placing new activity data in the string (enlarging current activityDailyRecord, or placing a new activityDailyRecord) that replaces older activity data, activityPointerOldestDayRecord must be updated to reflect the new location of the oldest complete day record, and activityPreviousRecordLength of this (new) oldest complete day record must be reset to 0.

2.18. CardDrivingLicenceInformation

Information, stored in a driver card, related to the card holder driver licence data (~~Annex~~ **Appendix 1C** requirement 259 and 284).

```
CardDrivingLicenceInformation ::= SEQUENCE {
    drivingLicenceIssuingAuthority      Name,
    drivingLicenceIssuingNation        NationNationNumeric,
    drivingLicenceNumber                IA5String(SIZE(16))
}
```

drivingLicenceIssuingAuthority is the authority responsible for issuing the driving licence.

drivingLicenceIssuingNation is the nationality of the authority that issued the driving licence.

drivingLicenceNumber is the number of the driving licence.

2.19. CardEventData

Generation 1

Information, stored in a driver or workshop card, related to the events associated with the card holder (~~Annex~~ **Appendix 1C** requirements 260 ~~285,318~~ and ~~344~~ **318**).

```
CardEventData ::= SEQUENCE SIZE(6) OF {
    cardEventRecords                SET SIZE(NoOfEventsPerType) OF CardEventRecord
}
```

CardEventData is a sequence, ordered by ascending value of EventFaultType, of cardEventRecords (except security breach attempts related records which are gathered in the last set of the sequence).

cardEventRecords is a set of event records of a given event type (or category for security breach attempts events).

Generation 2

Information, stored in a driver or workshop card, related to the events associated with the card holder (Appendix 1C requirements 285 and 341).

```
CardEventData ::= SEQUENCE SIZE(11) OF {
cardEventRecords      SET SIZE(NoOfEventsPerType) OF CardEventRecord
}
```

CardEventData is a sequence, ordered by ascending value of **EventFaultType**, of **cardEventRecords** (except security breach attempts related records which are gathered in the last set of the sequence).

cardEventRecords is a set of event records of a given event type (or category for security breach attempts events).

2.20. CardEventRecord

Information, stored in a driver or a workshop card, related to an event associated to the card holder (Appendix 1C requirements 261, 286, 318 and 341).

```
CardEventRecord ::= SEQUENCE {
eventType              EventFaultType,
eventBeginTime         TimeReal,
eventEndTime           TimeReal,
eventVehicleRegistration VehicleRegistrationIdentification
}
```

eventType is the type of the event.

eventBeginTime is the date and time of beginning of event.

eventEndTime is the date and time of end of event.

eventVehicleRegistration is the VRN and registering ~~Member State~~ **Contracting Party** of vehicle in which the event happened.

2.21. CardFaultData

Information, stored in a driver or a workshop card, related to the faults associated to the card holder (~~Annex~~ **Appendix** 1C requirements 263, 288, 318, and 341).

```
CardFaultData ::= SEQUENCE SIZE(2) OF {
cardFaultRecords      SET SIZE(NoOfFaultsPerType) OF CardFaultRecord
}
```

CardFaultData is a sequence of ~~Recording Equipment~~ **Control device** faults set of records followed by card faults set of records.

cardFaultRecords is a set of fault records of a given fault category (~~Recording Equipment~~ **Control device** or card).

2.22. CardFaultRecord

Information, stored in a driver or a workshop card, related to a fault associated to the card holder (~~Annex~~ **Appendix** 1C requirement 264, 289, 318, and 341).

```
CardFaultRecord ::= SEQUENCE {
    faultType          EventFaultType,
    faultBegin         TimeTimeReal,
    faultEndTime       TimeReal,
    faultVehicleRegistration VehicleRegistrationIdentification
}
```

faultType is the type of the fault.

faultBeginTime is the date and time of beginning of fault.

faultEndTime is the date and time of end of fault.

faultVehicleRegistration is the VRN and registering ~~Member State~~ **Contracting Party** of vehicle in which the fault happened.

2.23. CardIccIdentification

Information, stored in a card, related to the identification of the integrated circuit (IC) card (~~Annex~~ **Appendix** 1C requirement 248).

```
CardIccIdentification ::= SEQUENCE {
    clockStop          OCTET STRING (SIZE(1)),
    cardExtendedSerialNumber ExtendedSerialNumber,
    cardApprovalNumber CardApprovalNumber,
    cardPersonaliserID ManufacturerCode,
    embedderIcAssemblerId EmbedderIcAssemblerId,
    icIdentifier        OCTET STRING (SIZE(2))
}
```

clockStop is the Clockstop mode as defined in **Sub**-appendix 2.

cardExtendedSerialNumber is the IC card unique serial number as further specified by the ExtendedSerialNumber data type.

cardApprovalNumber is the type approval number of the card.

cardPersonaliserID is the card personaliser ID encoded as ManufacturerCode.

embedderIcAssemblerId provides information about the embedder/IC assembler.

icIdentifier is the Identifier of the IC on the card and its IC manufacturer as defined in ISO/IEC 7816-6.

2.24. CardIdentification

Information, stored in a card, related to the identification of the card (~~Annex~~ **Appendix 1C** requirements 255, 280, 310, 333, 359, 365, 371, and 377).

```
CardIdentification ::= SEQUENCE {
    cardIssuingMemberState      NationNumeric,
    cardNumber                  CardNumber,
    cardIssuingAuthorityName    Name,
    cardIssueDate               TimeReal,
    cardValidityBegin           TimeReal,
    cardExpiryDate              TimeReal
}
```

cardIssuingMemberState is the code of the ~~Member State~~ **Contracting Party** issuing the card.

cardNumber is the card number of the card.

cardIssuingAuthorityName is the name of the authority having issued the ~~Card~~ **card**.

cardIssueDate is the issue date of the ~~Card~~ **card** to the current holder.

cardValidityBegin is the first date of validity of the card.

cardExpiryDate is the date when the validity of the card ends.

2.24a. CardLoadTypeEntries

Generation 2, version 2:

Information, stored in a driver or workshop card, related to the load type entries when the card is inserted in a vehicle unit (**Appendix 1C** requirements 306j and 356j).

```
CardLoadTypeEntries ::= SEQUENCE {
    loadTypeEntryPointerNewestRecord    INTEGER(0..NoOfLoadTypeEntryRecords -1),
    cardLoadTypeEntryRecords           SET SIZE(NoOfLoadTypeEntryRecords) OF
                                        CardLoadTypeEntryRecord
}
```

loadTypeEntryPointerNewestRecord is the index of the last updated card load type entry record.

Value assignment: number corresponding to the numerator of the card load type entry record, beginning with '0' for the first occurrence of the card load type entry record in the structure.

cardLoadTypeEntryRecords is the set of records containing the date and time of the entry and the load type entered.

2.24b. CardLoadTypeEntryRecord

Generation 2, version 2:

Information, stored in a driver or workshop card, related to the load type changes entered when the card is inserted in a vehicle unit (Appendix 1C requirements 306i and 356i).

```
CardLoadTypeEntryRecord ::= SEQUENCE {
    timeStamp                TimeReal,
    loadTypeEntered          LoadType
}
```

timeStamp is the date and time when the load type was entered.

loadTypeEntered is the load type entered.

2.24c. CardLoadUnloadOperations

Generation 2, version 2:

Information, stored in a driver or workshop card, related to load/unload operations of the vehicle (Appendix 1C requirements 306h and 356h).

```
CardLoadUnloadOperations ::= SEQUENCE {
    loadUnloadPointerNewestRecord    INTEGER(0..NoOfLoadUnloadRecords -1),
    cardLoadUnloadRecords            SET SIZE(NoOfLoadUnloadRecords) OF
                                     CardLoadUnloadRecord
}
```

loadUnloadPointerNewestRecord is the index of the last updated card load/unload record.

Value assignment: is the number corresponding to the numerator of the card load/unload record, beginning with '0' for the first occurrence of the card load/unload record in the structure.

cardLoadUnloadRecords is the set of records containing the indication of the type of operation performed (load, unload, or simultaneous load and unload), the date and time the load/unload operation has been entered, information about the position of the vehicle, and the vehicle odometer value.

2.24d. CardLoadUnloadRecord

Generation 2, version 2:

Information, stored in a driver or workshop card, related to load/unload operations of the vehicle (Appendix 1C requirements 306g and 356g).

```
CardLoadUnloadRecord ::= SEQUENCE {
    timeStamp                TimeReal,
    operationType            OperationType,
    gnssPlaceAuthRecord      GNSSPlaceAuthRecord,
    vehicleOdometerValue     OdometerShort
}
```

timeStamp is the date and time at the beginning of the load/unload operation.

operationType is the type of operation entered (load, unload, or simultaneous load/unload).

gnssPlaceAuthRecord contains information related to the position of the vehicle.

vehicleOdometerValue is the odometer value related to the beginning of the load/unload operation.

2.25. CardMACertificate

Generation 2:

Certificate of the card public key for mutual authentication with a VU. The structure of this certificate is specified in ~~Appendix~~ **Sub**-appendix 11.

CardMACertificate ::= Certificate

2.26. CardNumber

A card number as defined by definition g).

```
CardNumber ::= CHOICE {
SEQUENCE {
driverIdentification          IA5String(SIZE(14)),
cardReplacementIndex        CardReplacementIndex,
cardRenewalIndex            CardRenewalIndex
},
SEQUENCE {
ownerIdentification          IA5String(SIZE(13)),
cardConsecutiveIndex        CardConsecutiveIndex,
cardReplacementIndex        CardReplacementIndex,
cardRenewalIndex            CardRenewalIndex
}
}
```

driverIdentification is the unique identification of a driver in a ~~Member State~~ **Contracting Party**.

ownerIdentification is the unique identification of a company or a workshop or a control body within a ~~member state~~ **Contracting Party**.

cardConsecutiveIndex is the card consecutive index.

cardReplacementIndex is the card replacement index.

cardRenewalIndex is the card renewal index.

The first sequence of the choice is suitable to code a driver card number, the second sequence of the choice is suitable to code workshop, control, and company card numbers.

2.26a. CardPlaceAuthDailyWorkPeriod

Generation 2, version 2:

Information, stored in a driver or a workshop card, providing the authentication status of places where daily work periods begin and/or end (Appendix 1C requirements 306b and 356b).

```
CardPlaceAuthDailyWorkPeriod ::= SEQUENCE {
  placeAuthPointerNewestRecord    INTEGER(0 .. NoOfCardPlaceRecords-1),
  placeAuthStatusRecords          SET      SIZE(NoOfCardPlaceRecords)    OF
                                  PlaceAuthStatusRecord
}
```

placeAuthPointerNewestRecord is the index of the last updated place authentication status record.

Value assignment: Number corresponding to the numerator of the place authentication status record, beginning with '0' for the first occurrence of the place authentication status records in the structure.

placeAuthStatusRecords is the set of records containing the place authentication status of the places entered.

2.27. CardPlaceDailyWorkPeriod

Information, stored in a driver or a workshop card, related to the places where daily work periods begin and/or end (Annex Appendix 1C requirements 272, 297, 325, and 348).

```
CardPlaceDailyWorkPeriod ::= SEQUENCE {
  placePointerNewestRecord        INTEGER(0 .. NoOfCardPlaceRecords-1),
  placeRecords                    SET      SIZE(NoOfCardPlaceRecords)    OF
  PlaceRecord
}
```

placePointerNewestRecord is the index of the last updated place record.

Value assignment: Number corresponding to the numerator of the place record, beginning with '0' for the first occurrence of the place records in the structure.

placeRecords is the set of records containing the information related to the places entered.

2.28. CardPrivateKey

Generation 1:

The private key of a card.

CardPrivateKey ::= RSAKeyPrivateExponent

2.29. CardPublicKey

The public key of a card. CardPublicKey ::= PublicKey

2.30. CardRenewalIndex

A card renewal index (definition i)).

CardRenewalIndex ::= IA5String(SIZE(1))

Value assignment: (see this ~~Annex~~ **Appendix** chapter ~~VII~~7).

‘0’ First issue.

Order for increase: ‘0 , ... , 9 , A , ... , Z’

2.31. CardReplacementIndex

A card replacement index (definition j)).

CardReplacementIndex ::= IA5String(SIZE(1))

Value assignment: (see this ~~Annex~~ **Appendix** chapter VII).

‘0’ Original card.

Order for increase: ‘0 , ... , 9 , A , ... , Z’

2.32. CardSignCertificate

Generation 2:

Certificate of the card public key for signature. The structure of this certificate is specified in ~~Appendix~~ **Sub-appendix** 11.

CardSignCertificate ::= Certificate

2.33. CardSlotNumber

Code to distinguish between the two slots of a Vehicle Unit.

CardSlotNumber ::= INTEGER {

driverSlot (0),

co-driverSlot (1)

}

Value assignment: not further specified.

2.34. CardSlotsStatus

Code indicating the type of cards inserted in the two slots of the vehicle unit.

CardSlotsStatus ::= OCTET STRING (SIZE(1))

Value assignment – Octet Aligned : ‘ccccddd’B

‘cccc’B Identification of the type of card inserted in the co-driver slot,

‘ddd’B Identification of the type of card inserted in the driver slot,

with the following identification codes:

'0000'B	no card is inserted,
'0001'B	a driver card is inserted,
'0010'B	a workshop card is inserted,
'0011'B	a control card is inserted,
'0100'B	a company card is inserted.

2.35. CardSlotsStatusRecordArray

Generation 2:

The CardSlotsStatus plus metadata as used in the download protocol.

```
CardSlotsStatusRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF CardSlotsStatus
}
```

recordType denotes the type of the record (CardSlotsStatus). **Value Assignment:** See RecordType

recordSize is the size of the CardSlotsStatus in bytes.

noOfRecords is the number of records in the set records.

records is the set of CardSlotsStatus records.

2.36. CardStructureVersion

Code indicating the version of the implemented structure in a tachograph card.

```
CardStructureVersion ::= OCTET STRING (SIZE(2))
```

Value assignment: 'aabb'H:

'aa'H Index for changes of the structure.

'00'H for Generation 1 applications

'01'H for Generation 2 applications

'bb'H Index for changes concerning the use of the data elements defined for the structure given by the high byte.

'00'H for ~~this version of~~ Generation 1 applications

'00'H for ~~this version 1 of~~ Generation 2 applications

'01'H for version 2 of Generation 2 applications

2.37. CardVehicleRecord

Information, stored in a driver or workshop card, related to a period of use of a vehicle during a calendar day (~~Annex~~ **Appendix 1C** requirements 269, 294, 322, and 345).

Generation 1:

```
CardVehicleRecord ::= SEQUENCE {
vehicleOdometerBegin      OdometerShort,
vehicleOdometerEnd       OdometerShort,
vehicleFirstUse           TimeReal,
vehicleLastUse           TimeReal,
vehicleRegistration       VehicleRegistrationIdentification,
vuDataBlockCounter       VuDataBlockCounter
}
```

vehicleOdometerBegin is the vehicle odometer value at the beginning of the period of use of the vehicle.

vehicleOdometerEnd is the vehicle odometer value at the end of the period of use of the vehicle.

vehicleFirstUse is the date and time of the beginning of the period of use of the vehicle.

vehicleLastUse is the date and time of the end of the period of use of the vehicle.

vehicleRegistration is the VRN and the registering ~~Member State~~ **Contracting Party** of the vehicle.

vuDataBlockCounter is the value of the VuDataBlockCounter at last extraction of the period of use of the vehicle.

Generation 2:

```
CardVehicleRecord ::= SEQUENCE {
vehicleOdometerBegin      OdometerShort,
vehicleOdometerEnd       OdometerShort,
vehicleFirstUse           TimeReal,
vehicleLastUse           TimeReal,
vehicleRegistration       VehicleRegistrationIdentification,
vuDataBlockCounter       VuDataBlockCounter,
vehicleIdentificationNumber VehicleIdentificationNumber
}
```

In addition to generation 1 the following data element is used:

VehicleIdentificationNumber is the vehicle identification number referring to the vehicle as a whole.

2.38. CardVehiclesUsed

Information, stored in a driver or workshop card, related to the vehicles used by the card holder (~~Annex~~ **Appendix** 1C requirements 270, 295, 323, and 346).

```
CardVehiclesUsed ::= SEQUENCE {
vehiclePointerNewestRecord    INTEGER(0..NoOfCardVehicleRecords-1),
```

```

cardVehicleRecords          SET    SIZE(NoOfCardVehicleRecords)  OF
CardVehicleRecord
}

```

vehiclePointerNewestRecord is the index of the last updated vehicle record.

Value assignment: Number corresponding to the numerator of the vehicle record, beginning with '0' for the first occurrence of the vehicle records in the structure.

cardVehicleRecords is the set of records containing information on vehicles used.

2.39. CardVehicleUnitRecord

Generation 2:

~~Generation 2:~~

Information, stored in a driver or workshop card, related to a vehicle unit that was used (~~Annex~~ **Appendix** 1C requirement 303 and 351).

```

CardVehicleUnitRecord ::= SEQUENCE {
timeStamp                TimeReal,
manufacturerCode        ManufacturerCode,
deviceID                 INTEGER(0..255),
vuSoftwareVersion       VuSoftwareVersion
}

```

timeStamp is the beginning of the period of use of the vehicle unit (i.e. first card insertion in the vehicle unit for the period).

manufacturerCode identifies the manufacturer of the Vehicle Unit.

deviceID identifies the Vehicle Unit type of a manufacturer. The value is manufacturer specific.

vuSoftwareVersion is the software version number of the Vehicle Unit.

2.40. CardVehicleUnitsUsed

Generation 2:

Information, stored in a driver or workshop card, related to the vehicle units used by the card holder (~~Annex~~ **Appendix** 1C requirements ~~304~~ and 352).

```

CardVehicleUnitsUsed ::= SEQUENCE {
vehicleUnitPointerNewestRecord  INTEGER(0..NoOfCardVehicleUnitRecords-1),
cardVehicleUnitRecords          SET    SIZE(NoOfCardVehicleUnitRecords)  OF
CardVehicleUnitRecord
}

```

vehicleUnitPointerNewestRecord is the index of the last updated vehicle unit record.

Value assignment: Number corresponding to the numerator of the vehicle unit record, beginning with '0' for the first occurrence of the vehicle unit records in the structure.

cardVehicleUnitRecords is the set of records containing information on vehicle units used.

2.41. Certificate

The certificate of a public key issued by a Certification Authority.

Generation 1:

Certificate ::= OCTET STRING (SIZE(194))

Value assignment: digital signature with partial recovery of a Certificate-Content according to ~~Appendix~~ **Sub-appendix** 11 common security mechanisms: Signature (128 bytes) || Public Key remainder (58 bytes) || Certification Authority Reference (8 bytes).

Generation 2:

Certificate ::= OCTET STRING (SIZE(204..341))

Value assignment: See ~~Appendix~~ **Sub-appendix** 11

2.42. CertificateContent

Generation 1:

The (clear) content of the certificate of a public key according to ~~Appendix~~ **Sub-appendix** 11 common security mechanisms.

CertificateContent ::= SEQUENCE {

certificateProfileIdentifier INTEGER(0..255),

certificationAuthorityReference KeyIdentifier,

certificateHolderAuthorisation CertificateHolderAuthorisation,

certificateEndOfValidity TimeReal,

certificateHolderReference KeyIdentifier,

publicKey PublicKey

}

certificateProfileIdentifier is the version of the corresponding certificate.

Value assignment: '01h' for this version.

certificationAuthorityReference identifies the Certification Authority issuing the certificate. It also references the Public Key of this Certification Authority.

certificateHolderAuthorisation identifies the rights of the certificate holder.

certificateEndOfValidity is the date when the certificate expires administratively.

certificateHolderReference identifies the certificate holder. It also references his Public Key.

publicKey is the public key that is certified by this certificate.

2.43. CertificateHolderAuthorisation

Identification of the rights of a certificate holder.

```
CertificateHolderAuthorisation ::= SEQUENCE {
tachographApplicationID      OCTET STRING(SIZE(6))
equipmentType                 EquipmentType
}
```

Generation 1:

tachographApplicationID is the application identifier for the tachograph application.

Value assignment: 'FFh' '54h' '41h' '43h' '48h' '4Fh'. This AID is a proprietary non registered application identifier in accordance with ISO/IEC 7816-5.

equipmentType is the identification of the type of equipment to which the certificate is intended.

Value assignment: in accordance with EquipmentType data type. **0** if certificate is the one of a ~~Member State~~ **Contracting Party**.

Generation 2:

tachographApplicationID denotes the 6 most significant bytes of the generation 2 tachograph card application identifier (AID). The AID for the tachograph card application is specified in chapter ~~6.2~~ **6.2**.

Value assignment: 'FF 53 4D 52 44 54'.

equipmentType is the identification of the type of equipment as specified for generation 2 to which the certificate is intended.

Value assignment: in accordance with EquipmentType data type.

2.44. CertificateRequestID

Unique identification of a certificate request. It can also be used as a Vehicle Unit Public Key Identifier if the serial number of the vehicle Unit to which the key is intended is not known at certificate generation time.

```
CertificateRequestID ::= SEQUENCE{
requestSerialNumber requestSerialNumber    INTEGER(0..232-1),
requestMonthYear           BCDString(SIZE(2)),
crIdentifier               OCTET STRING(SIZE(1)),
manufacturerCode          ManufacturerCode
}
```

requestSerialNumber is a serial number for the certificate request, unique for the manufacturer and the month below.

requestMonthYear is the identification of the month and the year of the certificate request.

Value assignment: BCD coding of Month (two digits) and Year (two last digits).

crIdentifier: is an identifier to distinguish a certificate request from an extended serial number.

Value assignment: 'FFh'.

manufacturerCode: is the numerical code of the manufacturer requesting the certificate.

2.45. CertificationAuthorityKID

Identifier of the Public Key of a Certification Authority (a ~~Member State Contracting Party~~ or the ~~European~~ **Root** Certification Authority).

```
CertificationAuthorityKID ::= SEQUENCE{
nationNumeric                NationNumeric,
nationAlpha                  NationAlpha,
keySerialNumber              INTEGER(0..255),
additionalInfo                OCTET STRING(SIZE(2)),
caIdentifier                  OCTET STRING(SIZE(1))
}
```

nationNumeric is the numerical nation code of the Certification Authority.

nationAlpha is the alphanumerical nation code of the Certification Authority.

keySerialNumber is a serial number to distinguish the different keys of the Certification Authority in the case keys are changed.

additionalInfo is a two byte field for additional coding (Certification Authority specific).

caIdentifier is an identifier to distinguish a Certification Authority Key Identifier from other Key Identifiers.

Value assignment: '01h'.

2.46. CompanyActivityData

Information, stored in a company card, related to activities performed with the card (~~Annex~~ **Appendix** 1C requirement 373 and 379).

```
CompanyActivityData ::= SEQUENCE {
companyPointerNewestRecord    INTEGER(0..NoOfCompanyActivityRecords-1),
companyActivityRecords        SET SIZE(NoOfCompanyActivityRecords) OF
companyActivityRecord         SEQUENCE {
companyActivityType           CompanyActivityType,
companyActivityTime           TimeReal,
cardNumberInformation         FullCardNumber,
vehicleRegistrationInformation VehicleRegistrationIdentification,
downloadPeriodBegin           TimeReal,
downloadPeriodEnd             TimeReal
}
}
```

companyPointerNewestRecord is the index of the last updated companyActivityRecord.

Value assignment: Number corresponding to the numerator of the company activity record, beginning with '0' for the first occurrence of the company activity record in the structure.

companyActivityRecords is the set of all company activity records.

companyActivityRecord is the sequence of information related to one company activity.

companyActivityType is the type of the company activity.

companyActivityTime is the date and time of the company activity.

cardNumberInformation is the card number and the card issuing ~~Member State~~ **Contracting Party** of the card downloaded, if any.

vehicleRegistrationInformation is the VRN and registering ~~Member State~~ **Contracting Party** of the vehicle downloaded or locked in or out.

downloadPeriodBegin and **downloadPeriodEnd** is the period downloaded from the VU, if any.

2.47. **CompanyActivityType**

Code indicating an activity carried out by a company using its company card.

```
CompanyActivityType ::= INTEGER {  
card downloading (1),  
VU downloading (2),  
VU lock-in (3),  
VU lock-out (4)  
}
```

2.48. **CompanyCardApplicationIdentification**

Information, stored in a company card related to the identification of the application of the card (~~Annex~~ **Appendix 1C** requirement 369 and 375).

```
CompanyCardApplicationIdentification ::= SEQUENCE {  
typeOfTachographCardId EquipmentType,  
cardStructureVersion CardStructureVersion,  
noOfCompanyActivityRecords NoOfCompanyActivityRecords  
}
```

typeOfTachographCardId is specifying the implemented type of card.

cardStructureVersion is specifying the the version of the structure that is implemented in the card.

noOfCompanyActivityRecords is the number of company activity records the card can store.

2.48a. **CompanyCardApplicationIdentificationV2**

Generation 2, version 2:

Information, stored in a company card related to the identification of the application of the card (Appendix 1C requirement 375a).

```
CompanyCardApplicationIdentificationV2 ::= SEQUENCE {
```


lengthOfFollowingData **LengthOfFollowingData,**
vuConfigurationLengthRange **VuConfigurationLengthRange**
 }

lengthOfFollowingData is the number of bytes following in the record.

vuConfigurationLengthRange is the number of bytes in a tachograph card, available to store VU configurations.

2.49. CompanyCardHolderIdentification

Information, stored in a company card, related to the cardholder identification (~~Annex~~ **Appendix 1C** requirement 372 and 378).

```
CompanyCardHolderIdentification ::= SEQUENCE {
  companyName                            Name,
  companyAddress                        Address,
  cardHolderPreferredLanguage        Language
}
```

companyName is the name of the holder company.

companyAddress is the address of the holder company.

cardHolderPreferredLanguage is the preferred language of the card holder.

2.50. ControlCardApplicationIdentification

Information, stored in a control card related to the identification of the application of the card (~~Annex~~ **Appendix 1C** requirement 357 and 363).

```
ControlCardApplicationIdentification ::= SEQUENCE {
  typeOfTachographCardId            EquipmentType,
  cardStructureVersion               CardStructureVersion,
  noOfControlActivityRecords        NoOfControlActivityRecords
}
```

typeOfTachographCardId is specifying the implemented type of card.

cardStructureVersion is specifying the version of the structure that is implemented in the card.

noOfControlActivityRecords is the number of control activity records the card can store.

2.50a. ControlCardApplicationIdentificationV2

Generation 2, version 2:

Information, stored in a control card related to the identification of the application of the card (**Appendix 1C** requirement 363a).

```
ControlCardApplicationIdentificationV2 ::= SEQUENCE {
```

lengthOfFollowingData **LengthOfFollowingData,**
vuConfigurationLengthRange **VuConfigurationLengthRange**
 }

lengthOfFollowingData is the number of bytes following in the record.

vuConfigurationLengthRange is the number of bytes in a tachograph card, available to store VU configurations.

2.51. ControlCardControlActivityData

Information, stored in a control card, related to control activity performed with the card (~~Annex~~ **Appendix 1C** requirement 361 and 367).

```
ControlCardControlActivityData ::= SEQUENCE {
controlPointerNewestRecord        INTEGER(0.. NoOfControlActivityRecords-1),
controlActivityRecords            SET SIZE(NoOfControlActivityRecords) OF
controlActivityRecord            SEQUENCE {
controlType                        ControlType,
controlTime                        TimeReal,
controlledCardNumber              FullCardNumber,
controlledVehicleRegistration     VehicleRegistrationIdentification,
controlDownloadPeriodBegin        TimeReal,
controlDownloadPeriodEnd          TimeReal
                                      }
}
```

controlPointerNewestRecord is the index of the last updated control activity record.

Value assignment: Number corresponding to the numerator of the control activity record, beginning with '0' for the first occurrence of the control activity record in the structure.

controlActivityRecords is the set of all control activity records.

controlActivityRecord is the sequence of information related to one control.

controlType is the type of the control.

controlTime is the date and time of the control.

controlledCardNumber is the card number and the card issuing ~~Member State~~ **Contracting Party** of the card controlled.

controlledVehicleRegistration is the VRN and registering ~~Member State~~ **Contracting Party** of the vehicle in which the control happened.

controlDownloadPeriodBegin and **controlDownloadPeriodEnd** is the period eventually downloaded.

2.52. ControlCardHolderIdentification

Information, stored in a control card, related to the identification of the cardholder (~~Annex~~ **Appendix 1C** requirement 360 and 366).

```
ControlCardHolderIdentification ::= SEQUENCE {
controlBodyName          Name,
controlBodyAddress       Address,
cardHolderName           HolderName,
cardHolderPreferredLanguage Language
}
```

controlBodyName is the name of the control body of the card holder.

controlBodyAddress is the address of the control body of the card holder.

cardHolderName is the name and first name(s) of the holder of the Control Card.

cardHolderPreferredLanguage is the preferred language of the card holder.

2.53. ControlType

Code indicating the activities carried out during a control. This data type is related to ~~Annex~~ **Appendix 1C** requirements 126, 274, 299, 327, and 350.

```
ControlType ::= OCTET STRING (SIZE(1))
```

Generation 1:

Value assignment – Octet aligned : ‘cvpdxxxx’B (8 bits)

‘c’B card downloading:

‘0’B: card not downloaded during this control activity,

‘1’B: card downloaded during this control activity

‘v’B VU downloading:

‘0’B: VU not downloaded during this control activity,

‘1’B: VU downloaded during this control activity

‘p’B printing:

‘0’B: no printing done during this control activity,

‘1’B: printing done during this control activity

‘d’B display:

‘0’B: no display used during this control activity,

‘1’B: display used during this control activity

‘xxxx’B Not used.

Generation 2:

Value assignment – Octet aligned : ‘cvpdexxx’B (8 bits)

‘c’B card downloading:

- '0'B: card not downloaded during this control activity,
- '1'B: card downloaded during this control activity
- 'v'B VU downloading:
 - '0'B: VU not downloaded during this control activity,
 - '1'B: VU downloaded during this control activity
- 'p'B printing:
 - '0'B: no printing done during this control activity,
 - '1'B: printing done during this control activity
- 'd'B display:
 - '0'B: no display used during this control activity,
 - '1'B: display used during this control activity
- 'e'B roadside calibration checking:
 - '0'B: calibration parameters not checked during this control activity,
 - '1'B: calibration parameters checked during this control activity
- 'xxx'B ~~RFU~~ **Reserved for future use.**

2.54. **CurrentDateTime**

The current date and time of the ~~recording equipment~~ **control device**.

CurrentDateTime ::= TimeReal

Value assignment: not further specified.

2.55. **CurrentDateTimeRecordArray**

Generation 2:

The current date and time plus metadata as used in the download protocol.

CurrentDateTimeRecordArray ::= SEQUENCE {

recordType RecordType,

recordSize INTEGER(1..65535),

noOfRecords INTEGER(0..65535),

records SET SIZE(noOfRecords) OF CurrentDateTime

}

recordType denotes the type of the record (CurrentDateTime). **Value Assignment:** See RecordType

recordSize is the size of the CurrentDateTime in bytes.

noOfRecords is the number of records in the set records.

records is a set of current date and time records.

2.56. DailyPresenceCounter

Counter, stored in a driver or workshop card, increased by one for each calendar day the card has been inserted in a VU. This data type is related to ~~Annex~~ **Appendix 1C** requirements 266, 299, 320, and 343.

DailyPresenceCounter ::= BCDString(SIZE(2))

Value assignment: Consecutive Number with maximum value = 9 999, starting again with 0. At the time of first issuing of the card the number is set to 0.

2.57. Datef

Date expressed in a readily printable numeric format.

Datef ::= SEQUENCE {

year	BCDString(SIZE(2)),
month	BCDString(SIZE(1)),
day	BCDString(SIZE(1))
}	

Value assignment:

yyyy	Year
mm	Month
dd	Day

'00000000'H denotes explicitly no date.

2.58. DateOfDayDownloaded

Generation 2:

The date and time of the download.

DateOfDayDownloaded ::= TimeReal

Value assignment: not further specified.

2.59. DateOfDayDownloadedRecordArray

Generation 2:

The date and time of the download plus metadata as used in the download protocol.

DateOfDayDownloadedRecordArray ::= SEQUENCE {

recordType	RecordType,
recordSize	INTEGER(1..65535),
noOfRecords	INTEGER(0..65535),
records	SET SIZE(noOfRecords) OF DateOfDayDownloaded
}	

recordType denotes the type of the record (DateOfDayDownloaded). **Value Assignment:**
See RecordType

recordSize is the size of the CurrentDateTime in bytes.

noOfRecords is the number of records in the set records.

records is the set of date and time of the download records.

2.60. Distance

A distance travelled (result of the calculation of the difference between two vehicle's odometer values in kilometers).

Distance ::= INTEGER(0..2¹⁶-1)

Value assignment: Unsigned binary. Value in km in the operational range 0 to 9 999 km.

2.60a. DownloadInterfaceVersion

Generation 2, version 2:

Code indicating the version of the download interface of a vehicle unit.

DownloadInterfaceVersion ::= OCTET STRING (SIZE(2))

Value assignment: 'aabb'H:

'aa'H '00'H: not used,

'01'H: Generation 2 vehicle unit,

'bb'H '00'H: not used,

'01'H: version 2 of Generation 2 vehicle unit.

2.61. DriverCardApplicationIdentification

Information, stored in a driver card related to the identification of the application of the card (~~Annex~~ **Appendix 1C** requirement 253 and 278).

Generation 1:

```
DriverCardApplicationIdentification ::= SEQUENCE {
  typeOfTachographCardId      EquipmentType,
  cardStructureVersion         CardStructureVersion,
  noOfEventsPerType           NoOfEventsPerType,
  noOfFaultsPerType           NoOfFaultsPerType,
  activityStructureLength      CardActivityLengthRange,
  noOfCardVehicleRecords      NoOfCardVehicleRecords,
  noOfCardPlaceRecords        NoOfCardPlaceRecords
}
```

typeOfTachographCardId is specifying the implemented type of card.

cardStructureVersion is specifying the the version of the structure that is implemented in the card.

noOfEventsPerType is the number of events per type of event the card can record.

noOfFaultsPerType is the number of faults per type of fault the card can record.

activityStructureLength indicates the number of bytes available for storing activity records.

noOfCardVehicleRecords is the number of vehicle records the card can contain.

noOfCardPlaceRecords is the number of places the card can record.

Generation 2:

```

DriverCardApplicationIdentification ::= SEQUENCE {
typeOfTachographCardId           EquipmentType,
cardStructureVersion              CardStructureVersion,
noOfEventsPerType                 NoOfEventsPerType,
noOfFaultsPerType                 NoOfFaultsPerType,
activityStructureLength           CardActivityLengthRange,
noOfCardVehicleRecords           NoOfCardVehicleRecords,
noOfCardPlaceRecords             NoOfCardPlaceRecords,
noOfGNSSCDRecords             NoOfGNSSCDRecords,
noOfGNSSADRecords             NoOfGNSSADRecords,
noOfSpecificConditionRecords     NoOfSpecificConditionRecords
noOfCardVehicleUnitRecords   NoOfCardVehicleUnitRecords
}

```

In addition to generation 1 the following data elements are used:

~~noOfGNSSCDRecords~~ **noOfGNSSADRecords** is the number of GNSS ~~continuous~~ **accumulated** driving records the card can store.

noOfSpecificConditionRecords is the number of specific condition records the card can store.

noOfCardVehicleUnitRecords is the number of vehicle units used records the card can store.

2.61a. DriverCardApplicationIdentificationV2

Generation 2, version 2:

Information, stored in a driver card related to the identification of the application of the card (Appendix 1C requirement 278a).

```

DriverCardApplicationIdentificationV2 ::= SEQUENCE {
lengthOfFollowingData             LengthOfFollowingData,
noOfBorderCrossingRecords        NoOfBorderCrossingRecords,
noOfLoadUnloadRecords           NoOfLoadUnloadRecords,
noOfLoadTypeEntryRecords        NoOfLoadTypeEntryRecords,
vuConfigurationLengthRange       VuConfigurationLengthRange
}

```

lengthOfFollowingData is the number of bytes following in the record.

noOfBorderCrossingRecords is the number of border crossing records the driver card can store.

noOfLoadUnloadRecords is the number of load/unload records the driver card can store.

noOfLoadTypeEntryRecords is the number of load type entry records the driver card can store.

vuConfigurationLengthRange is the number of bytes in a tachograph card, available to store VU configurations.

2.62. DriverCardHolderIdentification

Information, stored in a driver card, related to the identification of the cardholder (~~Annex~~ **Appendix 1C** requirement 256 and 281).

```
DriverCardHolderIdentification ::= SEQUENCE {
cardHolderName          HolderName,
cardHolderBirthDate     Datef,
cardHolderPreferredLanguage Language
}
```

cardHolderName is the name and first name(s) of the holder of the Driver Card.

cardHolderBirthDate is the date of birth of the holder of the Driver Card.

cardHolderPreferredLanguage is the preferred language of the card holder.

2.63. DSRCSecurityData

Generation 2:

For the definition of this data type, see Sub-appendix 11.

○ — DSRCSecurityData

~~Generation 2:~~

~~The plain text information and the MAC to be transmitted via DSRC from the tachograph to the Remote Interrogator (RI), see Appendix 11 Part B chapter 13.63 Reserved for details.~~ **future use**


```

DSRCSecurityData ::= SEQUENCE {
tagLenthPlainText _____ OCTET STRING(SIZE(2)),
currentDateTime _____ CurrentDateTime,
counter _____ INTEGER(0..224-1),
vuSerialNumber _____ VuSerialNumber,
dSRCKMKVersionNumber _____ INTEGER(SIZE(1)),
tagLengthMac _____ OCTET STRING(SIZE(2)),
mac _____ MAC
+

```

~~**tagLength** is part of the DER TLV encoding and shall be set to '81 10' (see Appendix 11 Part B chapter 13).~~

~~**currentDateTime** is the current date and time of the vehicle unit.~~

~~**counter** enumerates the RTM messages.~~

~~**vuSerialNumber** is the serial number of the vehicle unit.~~

~~**dSRCKMKVersionNumber** is the version number of the DSRC Master Key from which the VU specific DSRC keys were derived.~~

~~**tagLengthMac** is the tag and length of the MAC data object as part of the DER TLV encoding. The tag shall be set to '8E', the length shall encode the length of the MAC in octets (see Appendix 11 Part B chapter 13).~~

~~**mac** is the MAC calculated over the RTM message (see Appendix 11 Part B chapter 13).~~

2.64. EGFCertificate

Generation 2:

Certificate of the external GNSS facility public key for mutual authentication with a VU. The structure of this certificate is specified in **Appendix Sub-appendix 11**.

EGFCertificate ::= Certificate

2.65. EmbedderIcAssemblerId

Provides information about the IC embedder.

```

EmbedderIcAssemblerId ::= SEQUENCE{
countryCode _____ IA5String(SIZE(2)),
moduleEmbedder _____ BCDString(SIZE(2)),
manufacturerInformation _____ OCTET STRING(SIZE(1))
}

```

countryCode is the 2 letter country code of the module embedder according to ISO 3166

moduleEmbedder identifies the module embedder

manufacturerInformation for manufacturer internal usage.

2.66. EntryTypeDailyWorkPeriod

Code to distinguish between begin and end for an entry of a daily work period place and condition of the entry.

Generation 1

```
EntryTypeDailyWorkPeriod ::= INTEGER {
Begin,related time = card insertion time or time of entry           (0),
End,related time = card withdrawal time or time of entry           (1),
Begin,related time manually entered (start time)                   (2),
End,related time manually entered (end of work period)             (3),
Begin,related time assumed by VU                                   (4),
End,related time assumed by VU                                   (5)
}
```

Value assignment: according to ISO/IEC8824-1.

Generation 2

```
EntryTypeDailyWorkPeriod ::= INTEGER {
Begin, related time = card insertion time or time of entry (0),
End, related time = card withdrawal time or time of entry ( 1),
Begin, related time manually entered (start time) (2),
End, related time manually entered (end of work period) (3),
Begin,related time assumed by VU (4),
End,related time assumed by VU (5),
Begin,related time based on GNSS data (6),
Endrelated time based on GNSS data (7)
}
```

Value assignment: according to ISO/IEC8824-1.

2.67. EquipmentType

Code to distinguish different types of equipment for the tachograph application.

EquipmentType ::= INTEGER(0..255)

Generation 1:

```
--Reserved (0),
--Driver Card (1),
--Workshop Card (2),
--Control Card (3),
--Company Card (4),
```

--Manufacturing Card	(5),
--Vehicle Unit	(6),
--Motion Sensor	(7),
--RFU	(8..255)

Value assignment: According to ISO/IEC8824-1.

Value 0 is reserved for the purpose of designating a ~~Member State or Europe Contracting Party~~ **Root Authority** in the CHA field of certificates.

Generation 2:

The same values as in generation 1 are used with the following additions:

--GNSS Facility	(8),
--Remote Communication Module	(9),
--ITS interface module	(10),
--Plaque SealRecord	(11), -- may be used in
--M1/N1 Adapter SealRecord	(12), -- may be used in
-- European Root CA (ERCA)	(13),
-- Member State Contracting Party CA (MSCA)	(14),
--External GNSS connection	(15), -- may be used in SealRecord
--Unused	(16), -- used in SealDataVu
-- Driver Card (sSign)	(17), -- only to be used in the CHA field of a signing certificate
-- Workshop Card (sSign)	(18) , -- only to be used in the CHA field of a signing certificate
-- Vehicle Unit (sSign)	(19) , -- only to be used in the CHA field of a signing certificate
--RFU	(20..255)

Note 1: The generation 2 values for the Plaque, Adapter and the External GNSS connection as well as the generation 1 values for the Vehicle Unit and Motion Sensor may be used in SealRecord, i.e. if applicable.

Note 2: In the CardHolderAuthorisation (CHA) field of a generation 2 certificate, the values (1), (2) and (6) are to be interpreted as indicating a certificate for Mutual Authentication for the respective equipment type. For indicating the respective certificate for creating a digital signature, the values (17), (18) or (19) must be used.

2.68. EuropeanPublicKey

Generation 1:

The ~~European~~ **Root** public key.

EuropeanPublicKey ::= PublicKey

2.69. EventFaultRecordPurpose

Code explaining why an event or a fault has been recorded.

EventFaultRecordPurpose ::= OCTET STRING (SIZE(1))

Value assignment:

'00'H	one of the 10 most recent (or last) events or faults
'01'H	the longest event for one of the last 10 days of occurrence
'02'H	one of the 5 longest events over the last 365 days
'03'H	the last event for one of the last 10 days of occurrence
'04'H	the most serious event for one of the last 10 days of occurrence
'05'H	one of the 5 most serious events over the last 365 days
'06'H	the first event or fault having occurred after the last calibration
'07'H	an active/on-going event or fault
'08'H to '7F'H	RFU
'80'H to 'FF'H	manufacturer specific

2.70. EventFaultType

Code qualifying an event or a fault.

EventFaultType ::= OCTET STRING (SIZE(1))

Value assignment:

Generation 1:

'0x'H	General events,
'00'H	No further details,
'01'H	Insertion of a non valid card,
'02'H	Card conflict,
'03'H	Time overlap,
'04'H	Driving without an appropriate card,
'05'H	Card insertion while driving,
'06'H	Last card session not correctly closed,
'07'H	Over speeding,
'08'H	Power supply interruption,
'09'H	Motion data error,
'0A'H	Vehicle Motion Conflict,

'0B' to '0F'H		RFU,
'1x'H attempt events,		Vehicle unit related security breach
'10'H		No further details,
'11'H		Motion sensor authentication failure,
'12'H		Tachograph card authentication failure,
'13'H		Unauthorised change of motion sensor,
'14'H		Card data input integrity error
'15'H		Stored user data integrity error,
'16'H		Internal data transfer error,
'17'H		Unauthorised case opening,
'18'H		Hardware sabotage,
'19'H to '1F'H	RFU,	
'2x'H events		Sensor related security breach attempt
'20'H		No further details,
'21'H		Authentication failure,
'22'H		Stored data integrity error,
'23'H		Internal data transfer error,
'24'H		Unauthorised case opening,
'25'H		Hardware sabotage,
'26'H to '2F'H	RFU,	
'3x'H	Recording equipment Control device faults,	
'30'H		No further details,
'31'H		VU internal fault,
'32'H		Printer fault,
'33'H		Display fault,
'34'H		Downloading fault,
'35'H		Sensor fault,
'36'H to '3F'H		RFU,
'4x'H		Card faults,
'40'H		No further details,
'41'H to '4F'H		RFU,
'50'H to '7F'H		RFU,
'80'H to 'FF'H		Manufacturer specific.

Generation 2, **version 1**:

The same values as in generation 1 are used with the following additions

'0x'H	General events,
'00'H	No further details,
'01'H	Insertion of a non valid card,
'02'H	Card conflict,
'03'H	Time overlap,
'04'H	Driving without an appropriate card,
'05'H	Card insertion while driving,
'06'H	Last card session not correctly closed,
'07'H	Over speeding,
'08'H	Power supply interruption,
'09'H	Motion data error,
'0A'H	Vehicle Motion Conflict,
'0B'H	Time conflict (GNSS versus VU internal clock),
0C to '0C'H	Communication error with the remote communication facility,
'0D'H	Absence of position information from GNSS receiver,
'0E'H	Communication error with the external GNSS facility,
'0F'H	RFU,
'5x'H GNSS- '1x'H	Vehicle unit related faults security breach attempt events,
50H '10'H	No further details,
'51'H '11'H	Motion sensor authentication failure,
'12'H	Tachograph card authentication failure,
'13'H	Unauthorised change of motion sensor,
'14'H	Card data input integrity error
'15'H	Stored user data integrity error,
'16'H	Internal GNSS receiver fault data transfer error,
52'H	External GNSS receiver fault,
'53'H	External GNSS communication fault,
'54'H	No GNSS position data,
'55'H '17'H	Unauthorised case opening,
'18'H	Hardware sabotage,
'19'H	Tamper detection of GNSS,
'56'H '1A'H	External GNSS facility authentication failure,

'57'H-1C'H to '5F'H-1F'H	External GNSS facility certificate expired, RFU,
'6x'H '2x'H	Sensor related security breach attempt events,
'20'H	No further details,
'21'H	Authentication failure,
'22'H	Stored data integrity error,
'23'H	Internal data transfer error,
'24'H	Unauthorised case opening,
'25'H	Hardware sabotage,
'26'H to '2F'H	RFU,
'3x'H	Control device faults,
'30'H	No further details,
'31'H	VU internal fault,
'32'H	Printer fault,
'33'H	Display fault,
'34'H	Downloading fault,
'35'H	Sensor fault,
'36'H	Internal GNSS receiver,
'37'H	External GNSS facility,
'38'H Module fault,	Remote communication facility Communication
'60'H	No further details
'61'H	Remote Communications Module fault
'62'H	Remote Communications Module communications fault facility
'63'H to '6F'H	RFU
'7x'H '39'H	ITS interface faults
'70'H-3A'H to '3F'H	RFU
'4x'H	Card faults,
'40'H	No further details,
'71'H '41F'H to '4F'H	RFU,
'50'H to '7F'H	RFU,
'80'H to 'FF'H	Manufacturer specific.

Generation 2, version 2:

- '0x'H General events,
- '00'H No further details,
- '01'H Insertion of a non valid card,
- '02'H Card conflict,
- '03'H Time overlap,
- '04'H Driving without an appropriate card,
- '05'H Card insertion while driving,
- '06'H Last card session not correctly closed,
- '07'H Over speeding,
- '08'H Power supply interruption,
- '09'H Motion data error,
- '0A'H Vehicle Motion Conflict,
- '0B'H Time conflict (GNSS versus VU internal clock),
- '0C'H Communication error with the remote communication facility,
- '0D'H Absence of position information from GNSS receiver,
- '0E'H Communication error with the external GNSS facility,
- '0F'H GNSS anomaly,

- '1x'H Vehicle unit related security breach attempt events,
- '10'H No further details,
- '11'H Motion sensor authentication failure,
- '12'H Tachograph card authentication failure,
- '13'H Unauthorised change of motion sensor,
- '14'H Card data input integrity error,
- '15'H Stored user data integrity error,
- '16'H Internal data transfer error,
- '17'H Unauthorised case opening,
- '18'H Hardware sabotage,
- '19'H Tamper detection of GNSS,
- '1A'H External GNSS facility authentication failure,
- '1B'H External GNSS facility certificate expired,
- '1C'H Inconsistency between motion data and stored driver activity data,
- '1D'H to '1F'H RFU,

- '2x'H Sensor related security breach attempt events,
- '20'H No further details,

- '21'H Authentication failure,
- '22'H Stored data integrity error,
- '23'H Internal data transfer error,
- '24'H Unauthorised case opening,
- '25'H Hardware sabotage,
- '26'H to '2F'H RFU,

- '3x'H Recording equipment faults,
- '30'H No further details,
- '31'H VU internal fault,
- '32'H Printer fault,
- '33'H Display fault,
- '34'H Downloading fault,
- '35'H Sensor fault,
- '36'H Internal GNSS receiver,
- '37'H External GNSS facility,
- '38'H Remote communication facility,
- '39'H ITS interface,
- '3A'H Internal Sensor Fault,
- to '3B'H to '3F'H RFU,

- '4x'H Card faults,
- '40'H No further details,
- '41'H to '4F'H RFU,

- '50'H to '7F'H RFU,

- '80'H to 'FF'H Manufacturer specific.

2.71. ExtendedSealIdentifier

Generation 2:

The extended seal identifier uniquely identifies a seal (~~Annex~~ **Appendix 1C** requirement 401).

```
ExtendedSealIdentifier ::= SEQUENCE{
manufacturerCode  IA5StringOCTET-STRING (SIZE(2)),
sealIdentifier     IA5StringOCTET-STRING (SIZE(68))
}
```

manufacturerCode is a code of the manufacturer of the seal. **Value assignment:** see database registration to be managed by the laboratory competent for interoperability tests (see <https://dtc.irc.ec.europa.eu>).

sealIdentifier is an identifier for the seal which is unique for the manufacturer. **Value assignment:** alpha-numeric number, unique in the manufacturer domain according to [ISO8859-1].

2.72. ExtendedSerialNumber

Unique identification of an equipment. It can also be used as an equipment Public Key Identifier.

Generation 1:

```
ExtendedSerialNumber ::= SEQUENCE{
serialNumber          INTEGER(0..232-1),
monthYear             BCDSString(SIZE(2)),
type                  OCTET STRING(SIZE(1)),
manufacturerCode     ManufacturerCode
}
```

serialNumber is a serial number for the equipment, unique for the manufacturer, the equipment's type and the month and year below.

monthYear is the identification of the month and the year of manufacturing (or of serial number assignment).

Value assignment: BCD coding of Month (two digits) and Year (two last digits).

type is an identifier of the type of equipment.

Value assignment: manufacturer specific, with 'FFh' reserved value.

manufacturerCode: is the numerical code identifying a manufacturer of type approved equipment.

Generation 2:

```
ExtendedSerialNumber ::= SEQUENCE{
serialNumber          INTEGER(0..232-1),
monthYear             BCDSString(SIZE(2)),
type                  EquipmentType,
manufacturerCode     ManufacturerCode
}
```

serialNumber see Generation 1

monthYear see Generation 1

type indicates the type of equipment

manufacturerCode: see Generation 1.

2.73. FullCardNumber

Code fully identifying a tachograph card.

```
FullCardNumber ::= SEQUENCE {
cardType                EquipmentType,
cardIssuingMemberState  NationNumeric,
cardNumber              CardNumber
}
```

cardType is the type of the tachograph card.

cardIssuingMemberState is the code of the ~~Member State~~ **Contracting Party** having issued the card.

cardNumber is the card number.

2.74. FullCardNumberAndGeneration

Generation 2:

Code fully identifying a tachograph card and its generation.

```
FullCardNumberAndGeneration ::= SEQUENCE {
fullCardNumber          FullCardNumber,
generation              Generation
}
```

fullcardNumber identifies the tachograph card.

generation indicates the generation of the tachograph card used.

2.75. Generation

Generation 2:

Indicates the generation of tachograph used.

```
Generation ::= INTEGER(0..255)
```

Value assignment:

'00'H	RFU
'01'H	Generation 1
'02'H	Generation 2
'03'H .. 'FF'H	RFU

2.76. GeoCoordinates

Generation 2:

The geo-coordinates are encoded as integers. These integers are multiples of the \pm DDMM.M encoding for the latitude and \pm DDDMM.M for the longitude. Here \pm DD respectively \pm DDD

denotes the degrees and MM.M the minutes. **Longitude and latitude of an unknown position shall be represented as Hex '7FFFFF' (Decimal 8388607).**

```
GeoCoordinates ::= SEQUENCE {
latitude          INTEGER(-90000..90001),
longitude        INTEGER(-180000..180001)
}
```

latitude is encoded as a multiple (factor 10) of the \pm DDMM.M representation.

longitude is encoded as a multiple (factor 10) of the \pm DDDMM.M representation.

2.77. GNSSAccuracy

Generation 2:

The accuracy of the GNSS position data (definition eee)). This accuracy is encoded as integer and is a multiple (factor 10) of the X.Y value provided by the GSA NMEA sentence.

```
GNSSAccuracy ::= INTEGER(1..100)
```

~~2.78 GNSSContinuousDriving~~

2.78. GNSSAccumulatedDriving

Generation 2:

Information, stored in a driver or workshop card, related to the GNSS position of the vehicle if the **continuous accumulated** driving time of the driver reaches a multiple of three hours (Annex Appendix 1C requirement 306 and 354).

```
GNSSContinuousDriving GNSSAccumulatedDriving ::= SEQUENCE {
gnssCDPointerNewestRecord gnssADPointerNewestRecord
INTEGER(0..NoOfGNSSADRecords-NoOfGNSSADRecords -1),
gnssContinuousDrivingRecords gnssAccumulatedDrivingRecords SET
SIZE(NoOfGNSSCDRecords-NoOfGNSSADRecords) OF GNSScontinuousDrivingRecord
GNSSAccumulatedDrivingRecord
}
```

~~gnssCDPointerNewestRecord~~~~gnssADPointerNewestRecord~~ is the index of the last updated GNSS **continuous-accumulated** driving record.

Value assignment Number is the number corresponding to the numerator of the GNSS **continuous accumulated** driving record, beginning with '0' for the first occurrence of the GNSS accumulated driving record in the structure.

~~gnssContinuousDrivingRecords~~~~gnssAccumulatedDrivingRecords~~ is the set of records containing the date and time the **continuous-accumulated** driving reaches a multiple of three hours and information on the position of the vehicle.

2.79. GNSSContinuousDrivingRecordGNSSAccumulatedDrivingRecord

Generation 2:

Information, stored in a driver or workshop card, related to the GNSS position of the vehicle if the ~~continuous~~ **accumulated** driving time of the driver reaches a multiple of three hours (Annex-Appendix 1C requirement 305 and 353).

```
GNSSContinuousDrivingRecordGNSSAccumulatedDrivingRecord ::= SEQUENCE {
timeStamp                               TimeReal,
gnssPlaceRecord                         GNSSPlaceRecord,
vehicleOdometerValue                    OdometerShort
}
```

timeStamp is the date and time when the ~~continuous~~ **accumulated** driving time of the card holder reaches a multiple of three hours.

gnssPlaceRecord contains information related to the position of the vehicle.

vehicleOdometerValue is the odometer value when the accumulated driving time reaches a multiple of three hours.

2.79a. GNSSAuthAccumulatedDriving

Generation 2, version 2:

Information, stored in a driver or workshop card, providing the authentication status of GNSS positions of the vehicle if the accumulated driving time reaches a multiple of three hours (Appendix 1C requirements 306d and 356d).

```
GNSSAuthAccumulatedDriving ::= SEQUENCE {
gnssAuthADPointerNewestRecord           INTEGER(0..NoOfGNSSADRecords -1),
gnssAuthStatusADRecords                 SET SIZE (NoOfGNSSADRecords) OF
GNSSAuthStatusADRecord
}
```

gnssAuthADPointerNewestRecord is the index of the last updated GNSS position authentication status record.

Value assignment is the number corresponding to the numerator of the GNSS position authentication status record, beginning with '0' for the first occurrence of the GNSS position authentication status record in the structure.

gnssAuthStatusADRecords is the set of records containing the date and time the accumulated driving reaches a multiple of three hours and the authentication status of the GNSS position.

2.79b. GNSSAuthStatusADRecord

Generation 2, version 2:

Information, stored in a driver or workshop card, providing the authentication status of a GNSS position of the vehicle if the accumulated driving time reaches a multiple of three hours (Appendix 1C requirements 306c and 356c). Other information related to the GNSS position itself is stored in another record (see 2.79 GNSSAccumulatedDrivingRecord).

```
GNSSAuthStatusADRecord ::= SEQUENCE {
```

timeStamp TimeReal,
authenticationStatus PositionAuthenticationStatus
}

timeStamp is the date and time when the accumulated driving time reaches a multiple of three hours (which is the same date and time as in the corresponding GNSSAccumulatedDrivingRecord).

authenticationStatus is the authentication status of the GNSS position when the accumulated driving time reaches a multiple of three hours.

2.79c. GNSSPlaceAuthRecord

Generation 2, version 2:

Information related to the GNSS position of the vehicle (Appendix 1C requirements 108, 109, 110, 296, 306a, 306c, 306e, 306g, 356a, 356c, 356e and 356g).

GNSSPlaceAuthRecord ::= SEQUENCE {

timeStamp TimeReal,
gnssAccuracy GNSSAccuracy,
geoCoordinates GeoCoordinates,
authenticationStatus PositionAuthenticationStatus
}

timeStamp is the date and time when the GNSS position of the vehicle was determined.

gnssAccuracy is the accuracy of the GNSS position data.

geoCoordinates is the recorded location using GNSS.

authenticationStatus is the authentication status of the GNSS position when it was determined.

2.80. GNSSPlaceRecord

Generation 2:

Information related to the GNSS position of the vehicle (~~Annex Appendix~~ 1C requirements 108, 109, 110, 296, 305, 347, and 353).

GNSSPlaceRecord ::= SEQUENCE {

timeStamp TimeReal,
gnssAccuracy GNSSAccuracy,
geoCoordinates GeoCoordinates
}

timeStamp is the date and time when the GNSS position of the vehicle was determined.

gnssAccuracy is the accuracy of the GNSS position data.

geoCoordinates is the recorded location using GNSS.

2.81. HighResOdometer

Odometer value of the vehicle: ~~Accumulated~~ **accumulated** distance travelled by the vehicle during its operation.

HighResOdometer ::= INTEGER (0..2³²-1)

Value assignment: Unsigned binary. Value in 1/200 km in the operating range 0 to 21 055 406 km.

2.82. HighResTripDistance

A distance travelled during all or part of a journey.

HighResTripDistance ::= INTEGER(0..2³²-1)

Value assignment: Unsigned binary. Value in 1/200 km in the operating range 0 to 21 055 406 km.

2.83. HolderName

The surname and first name(s) of a card holder.

```
HolderName ::= SEQUENCE {
holderSurname      Name,
holderFirstNames  Name
}
```

holderSurname is the surname (family name) of the holder. This surname does not include titles.

Value assignment: When a card is not personal, holderSurname contains the same information as companyName or workshopName or controlBodyName.

holderFirstNames is the first name(s) and initials of the holder.

2.84. Reserved for future useInternalGNSSReceiver

~~Generation 2:~~

~~Information if the GNSS receiver is internal or external to the vehicle unit. True means that the GNSS receiver is internal to the VU. False means that the GNSS receiver is external.~~

~~InternalGNSSReceiver ::= BOOLEAN~~

2.85. K-ConstantOfRecordingEquipment

Constant of the ~~recording equipment~~ **control device** (definition m)).

K-ConstantOfRecordingEquipment ::= INTEGER(0..2¹⁶-1)

Value assignment: Pulses per kilometer in the operating range 0 to 64 255 pulses/km.

2.86. KeyIdentifier

A unique identifier of a Public Key used to reference and select the key. It also identifies the holder of the key.

```
KeyIdentifier ::= CHOICE {  
    extendedSerialNumber      ExtendedSerialNumber,  
    certificateRequestID      CertificateRequestID,  
    certificationAuthorityKID  CertificationAuthorityKID  
}
```

The first choice is suitable to reference the public key of a Vehicle Unit, ~~or of~~, a tachograph card **or an external GNSS facility**.

The second choice is suitable to reference the public key of a Vehicle Unit (in ~~the case cases~~ **where** the serial number of the Vehicle Unit cannot be known at certificate generation time).

The third choice is suitable to reference the public key of a ~~Member State~~ **Contracting Party**.

2.87. KMWCKey

Generation 2:

AES key and its associated key version used for VU – Motion Sensor pairing. For details see ~~Appendix~~ **Sub-appendix 11**.

```
KMWCKey ::= SEQUENCE {  
    kMWCKey          AESKey,  
    keyVersion       INTEGER (SIZE(1))  
}
```

kMWCKey is the length of the AES key concatenated with the key which is used for VU – Motion Sensor pairing.

keyVersion denotes the key version of the AES key.

2.88. Language

Code identifying a language.

```
Language ::= IA5String(SIZE(2))
```

Value assignment: Two-letter lower-case coding according to ISO 639.

2.89. LastCardDownload

Date and time, stored on a driver card, of last card download (for other purposes than control) ~~Annex~~ **Appendix 1C** requirement 257 and 282. This date is updateable by a VU or any card reader.

```
LastCardDownload ::= TimeReal
```

Value assignment: not further specified.

2.89a. LengthOfFollowingData

Generation 2, version 2:

Length indicator for extensible records.

LengthOfFollowingData ::= INTEGER(0.. 2¹⁶-1)

Value assignment: See Sub-appendix 2.

2.90. LinkCertificate

Generation 2:

The link certificate between European Root CA key pairs.

LinkCertificate ::= Certificate

2.90a. LoadType

Generation 2, version 2:

Code identifying a load type entered.

LoadType ::= INTEGER(0..255)ç

Value assignment:

'00'H	Undefined load type,
'01'H	Goods,
'02'H	Passengers,
'03'H .. 'FF'H	RFU.

2.91. L-TyreCircumference

Effective circumference of the wheel tyres (definition u).

L-TyreCircumference ::= INTEGER(0.. 2¹⁶-1)

Value assignment: Unsigned binary, value in 1/8 mm in the operating range 0 to 8 031 mm.

2.92. MAC

Generation 2:

A cryptographic checksum of 8, 12 or 16 bytes length corresponding to the cipher suites specified in ~~Appendix~~ **Sub-appendix 11**.

MAC ::= CHOICE {

mac8	OCTET STRING (SIZE(8)),
mac12	OCTET STRING (SIZE(12)),
mac16	OCTET STRING (SIZE(16))

}

2.93. ManualInputFlag

Code identifying whether a cardholder has manually entered driver activities at card insertion or not (~~Annex Appendix~~ 1B requirement 081 and ~~Annex Appendix~~ 1C requirement 102).

```
ManualInputFlag ::= INTEGER {  
noEntry          (0)  
manualEntries    (1)  
}
```

Value assignment: not further specified.

2.94. ManufacturerCode

Code identifying a manufacturer of type approved equipment.

```
ManufacturerCode ::= INTEGER(0..255)
```

The laboratory competent for interoperability tests maintains and publishes the list of manufacturer codes on its web site (~~Annex Appendix~~ 1C requirement 454).

ManufacturerCodes are provisionally assigned to developers of tachograph equipment on application to the laboratory competent for interoperability tests.

2.95. ManufacturerSpecificEventFaultData

Generation 2:

Manufacturer specific error codes simplify the error analysis and maintenance of vehicle units.

```
ManufacturerSpecificEventFaultData ::= SEQUENCE {  
manufacturerCode          ManufacturerCode,  
manufacturerSpecificErrorCode  OCTET STRING(SIZE(3))  
}
```

manufacturerCode identifies the manufacturer of the Vehicle Unit.

manufacturerSpecificErrorCode is an error code specific to the manufacturer.

2.96. MemberStateCertificate

The certificate of the public key of a ~~member state~~ **Contracting Party** issued by the ~~European certification authority~~ **Root Certification Authority**.

```
MemberStateCertificate ::= Certificate
```

2.97. MemberStateCertificateRecordArray

Generation 2:

The ~~member state~~ **Contracting Party** certificate plus metadata as used in the download protocol.

```

MemberStateCertificateRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                  SET SIZE(noOfRecords) OF MemberStateCertificate
}

```

recordType denotes the type of the record (MemberStateCertificate). **Value Assignment:** See RecordType

recordSize is the size of the MemberStateCertificate in bytes.

noOfRecords is the number of records in the set records. The value shall be set to 1 as the certificates may have different lengths.

records is the set of ~~member state~~ **Contracting Party** certificates.

2.98. MemberStatePublicKey

Generation 1:

The public key of a ~~Member State~~ **Contracting Party**.

```
MemberStatePublicKey ::= PublicKey
```

2.99. Name

A name.

```

Name ::= SEQUENCE {
codePage                INTEGER (0..255),
name                   OCTET STRING (SIZE(35))
}

```

codePage specifies a character set defined in Chapter 4,

name is a name encoded using the specified character set.

2.100. NationAlpha

Alphabetic reference to a country shall be in accordance with the distinguishing signs used on vehicles in international traffic (United Nations Vienna Convention on Road Traffic, 1968).

```
NationAlpha ::= IA5String(SIZE(3))
```

The Nation Alpha and Numeric codes shall be held on a list maintained on the website of the laboratory appointed to carry out interoperability testing, as set out in ~~Annex~~ **Appendix 1C** requirement 440.

2.101. NationNumeric

Numerical reference to a country.

NationNumeric ::= INTEGER(0 .. 255)

Value assignment: see data type 2.100 (NationAlpha).

Any amendment or updating of the Nation Alpha or Numeric specification described in the above paragraph shall only be made out after the appointed laboratory has obtained the views of type approved digital and smart tachograph vehicle unit manufacturers.

2.101a. NoOfBorderCrossingRecords

Generation 2, version 2:

Number of border crossing records a driver or workshop card can store.

NoOfBorderCrossingRecords ::= INTEGER(0.. 2¹⁶-1)

Value assignment: see Sub-appendix 2.

2.102. NoOfCalibrationRecords

Number of calibration records, a workshop card can store.

Generation 1:

NoOfCalibrationRecords ::= INTEGER(0..255)

Value assignment: see ~~Appendix~~ **Sub-appendix 2.**

Generation 2:

NoOfCalibrationRecords ::= INTEGER(0..216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2.**

2.103. NoOfCalibrationsSinceDownload

Counter indicating the number of calibrations performed with a workshop card since its last download (~~Annex~~ **Appendix 1C** requirement 317 and 340).

NoOfCalibrationsSinceDownload ::= INTEGER(0..216-1)

Value assignment: Not specified further.

2.104. NoOfCardPlaceRecords

Number of place records a driver or workshop card can store.

Generation 1:

NoOfCardPlaceRecords ::= INTEGER(0..255)

Value assignment: see ~~Appendix~~ **Sub-appendix 2.**

Generation 2:

NoOfCardPlaceRecords ::= INTEGER(0..216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2.**

2.105. NoOfCardVehicleRecords

Number of vehicles used records a driver or workshop card can store.

NoOfCardVehicleRecords ::= INTEGER(0.. 216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.106. NoOfCardVehicleUnitRecords

Generation 2:

Number of vehicle units used records a driver or workshop card can store.

NoOfCardVehicleUnitRecords ::= INTEGER(0.. 216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.107. NoOfCompanyActivityRecords

Number of company activity records, a company card can store.

NoOfCompanyActivityRecords ::= INTEGER(0.. 216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.108. NoOfControlActivityRecords

Number of control activity records, a control card can store.

NoOfControlActivityRecords ::= INTEGER(0.. 216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.109. NoOfEventsPerType

Number of events per type of event a card can store.

NoOfEventsPerType ::= INTEGER(0..255)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.110. NoOfFaultsPerType

Number of faults per type of fault a card can store.

NoOfFaultsPerType ::= INTEGER(0..255)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

~~2.111. NoOfGNSSCDRecords~~

2.111. NoOfGNSSADRecords

Generation 2:

Number of GNSS ~~continuous~~ **accumulated** driving records a card can store.

~~NoOfGNSSCDRecords~~ **NoOfGNSSADRecords** ::= INTEGER(0..216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.111a. NoOfLoadUnloadRecords

Generation 2, version 2:

Number of load/unload records a card can store.

NoOfLoadUnloadRecords ::= INTEGER(0..2¹⁶-1)

Value assignment: see Sub-appendix 2.

2.112. NoOfSpecificConditionRecords

Generation 2:

Number of specific condition records a card can store.

NoOfSpecificConditionRecords ::= INTEGER(0..216-1)

Value assignment: see ~~Appendix~~ **Sub-appendix 2**.

2.112a. NoOfLoadTypeEntryRecords

Generation 2, version 2:

Number of load type entry records a driver or workshop card can store.

NoOfLoadTypeEntryRecords ::= INTEGER(0..2¹⁶-1)

Value assignment: see Sub-appendix 2.

2.113. OdometerShort

Odometer value of the vehicle in a short form.

OdometerShort ::= INTEGER(0..224-1)

Value assignment: Unsigned binary. Value in km in the operating range 0 to 9 999 999 km.

2.114. OdometerValueMidnight

The vehicle's odometer value at midnight on a given day (~~Annex Appendix~~ **1B** requirement 090 and ~~Annex Appendix~~ **1C** requirement 113).

OdometerValueMidnight ::= OdometerShort

Value assignment: not further specified.

2.114a. OperationType

Generation 2, version 2:

Code identifying a type of operation entered.

OperationType ::= INTEGER(0..255)

Value assignment:

'00'H	RFU,
'01'H	Load operation,
'02'H	Unload operation,
'03'H	Simultaneous load/unload operation,
'04'H .. 'FF'H	RFU.

2.115. OdometerValueMidnightRecordArray

Generation 2:

The OdometerValueMidnight plus metadata used in the download protocol.

```
OdometerValueMidnightRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET SIZE(noOfRecords) OF
OdometerValueMidnight
}
```

recordType denotes the type of the record (OdometerValueMidnight).

Value Assignment: See RecordType

recordSize is the size of the OdometerValueMidnight in bytes.

noOfRecords is the number of records in the set records.

records is the set of OdometerValueMidnight records.

2.116. OverspeedNumber

Number of over speeding events since the last over speeding control.

OverspeedNumber ::= INTEGER(0..255)

Value assignment: 0 means that no over speeding event has occurred since the last over speeding control , 1 means that one over speeding event has occurred since the last over speeding control ...255 means that 255 or more over speeding events have occurred since the last over speeding control.

2.116a. PlaceAuthRecord

Information related to a place where a daily work period begins or ends (Appendix 1C requirements 108, 271, 296, 324 and 347).

Generation 2, version 2:

PlaceAuthRecord ::= SEQUENCE {

entryTime	TimeReal,
entryTypeDailyWorkPeriod	EntryTypeDailyWorkPeriod,
dailyWorkPeriodCountry	NationNumeric,
dailyWorkPeriodRegion	RegionNumeric,
vehicleOdometerValue	OdometerShort,
entryGNSSPlaceAuthRecord	GNSSPlaceAuthRecord

}

entryTime is a date and time related to the entry.

entryTypeDailyWorkPeriod is the type of entry.

dailyWorkPeriodCountry is the country entered.

dailyWorkPeriodRegion is the region entered.

vehicleOdometerValue is the odometer value at the time of place entry.

entryGNSSPlaceAuthRecord is the recorded location, GNSS authentication status and time.

2.116b PlaceAuthStatusRecord

Generation 2, version 2:

Information, stored in a driver or workshop card, providing the authentication status of a place where a daily work period begins or ends (Appendix 1C requirements 306a and 356a). Other information related to the place itself is stored in another record (see 2.117 PlaceRecord).

PlaceAuthStatusRecord ::= SEQUENCE {

entryTime	TimeReal,
authenticationStatus	PositionAuthenticationStatus

}

entryTime is a date and time related to the entry (which is the same date and time as in the corresponding PlaceRecord).

authenticationStatus is the authentication status of the recorded GNSS position.

2.117. PlaceRecord

Information related to a place where a daily work period begins or ends (Annex Appendix 1C requirements 108, 271, 296, 324 and 347).

Generation 1:


```

PlaceRecord ::= SEQUENCE {
entryTime                TimeReal,
entryTypeDailyWorkPeriod EntryTypeDailyWorkPeriod,
dailyWorkPeriodCountry  NationNumeric,
dailyWorkPeriodRegion   RegionNumeric,
vehicleOdometerValue     OdometerShort
}

```

entryTime is a date and time related to the entry.

entryTypeDailyWorkPeriod is the type of entry.

dailyWorkPeriodCountry is the country entered.

dailyWorkPeriodRegion is the region entered.

vehicleOdometerValue is the odometer value at the time of place entry.

Generation 2:

```

PlaceRecord ::= SEQUENCE {
entryTime                TimeReal,
entryTypeDailyWorkPeriod EntryTypeDailyWorkPeriod,
dailyWorkPeriodCountry  NationNumeric,
dailyWorkPeriodRegion   RegionNumeric,
vehicleOdometerValue     OdometerShort,
entryGNSSPlaceRecord    GNSSPlaceRecord
}

```

In addition to Generation 1 the following component is used:

entryGNSSPlaceRecord is the recorded location and time.

2.117a PositionAuthenticationStatus

Generation 2, version 2:

PositionAuthenticationStatus ::= INTEGER(0..255)

Value assignment (see Sub-appendix 12):

'00'H Not Authenticated (see Sub-appendix 12, requirement GNS_39),

'01'H Authenticated (see Appendix 12, requirement GNS_39),

'02'H .. 'FF'H RFU.

2.118. PreviousVehicleInfo

Information related to the vehicle previously used by a driver when inserting his card in a vehicle unit (~~Annex~~ **Appendix 1B** requirement 081 and ~~Annex~~ **Appendix 1C** requirement 102).

Generation 1:

```
PreviousVehicleInfo ::= SEQUENCE {  
  vehicleRegistrationIdentification      VehicleRegistrationIdentification,  
  cardWithdrawalTime                   TimeReal  
}
```

vehicleRegistrationIdentification is the VRN and the registering ~~Member State Contracting~~ **Party** of the vehicle.

cardWithdrawalTime is the card withdrawal date and time.

Generation 2:

```
PreviousVehicleInfo ::= SEQUENCE {  
  vehicleRegistrationIdentification      VehicleRegistrationIdentification,  
  cardWithdrawalTime                   TimeReal,  
  vuGeneration                          Generation  
}
```

In addition to generation 1 the following data element is used:

vuGeneration identifies the generation of the vehicle unit.

2.119. **PublicKey**

Generation 1:

A public RSA key.

```
PublicKey ::= SEQUENCE {  
  rsaKeyModulus                        RSAKeyModulus,  
  rsaKeyPublicExponent                 RSAKeyPublicExponent  
}
```

rsaKeyModulus is the Modulus of the key pair.

rsaKeyPublicExponent is the public exponent of the key pair.

2.120. **RecordType**

Generation 2:

Reference to a record type. This data type is used in RecordArrays.

```
RecordType ::= OCTET STRING(SIZE(1))
```

Value assignment:

'01'H	ActivityChangeInfo,
'02'H	CardSlotsStatus,
'03'H	CurrentDateTime,
'04'H	MemberStateCertificate,

'05'H	OdometerValueMidnight,
'06'H	DateOfDayDownloaded,
'07'H	SensorPaired,
'08'H	Signature,
'09'H	SpecificConditionRecord,
'0A'H	VehicleIdentificationNumber,
'0B'H	VehicleRegistrationNumber,
'0C'H	VuCalibrationRecord,
'0D'H	VuCardIWRecord,
'0E'H	VuCardRecord,
'0F'H	VuCertificate,
'10'H	VuCompanyLocksRecord,
'11'H	VuControlActivityRecord,
'12'H	VuDetailedSpeedBlock,
'13'H	VuDownloadablePeriod,
'14'H	VuDownloadActivityData,
'15'H	VuEventRecord,
'16'H	VuGNSSCDRecord, VuGNSSADRecord,
'17'H	VuITSConsentRecord,
'18'H	VuFaultRecord,
'19'H	VuIdentification,
'1A'H	VuOverSpeedingControlData,
'1B'H	VuOverSpeedingEventRecord,
'1C'H	VuPlaceDailyWorkPeriodRecord,
'1D'H	VuTimeAdjustmentGNSSRecord,
'1E'H	VuTimeAdjustmentRecord,
'1F'H	VuPowerSupplyInterruptionRecord,
'20'H	SensorPairedRecord,
'21'H	SensorExternalGNSSCoupledRecord,
'22'H to '7F'H	VuBorderCrossingRecord RFU,
'2380'H to 'FF'H specific	VuLoadUnloadRecord , Manufacturer
'24'H	VehicleRegistrationIdentification,
'25'H to '7F'H	RFU,
'80'H to 'FF'H	Manufacturer specific.

2.121. RegionAlpha

Alphabetic reference to a region within a specified country.

RegionAlpha ::= IA5STRING(SIZE(3))

Generation 1:

Value assignment:

‘ ’ No information available,

Spain:

‘AN’	Andalucía,
‘AR’	Aragón,
‘AST’	Asturias,
‘C’	Cantabria,
‘CAT’	Cataluña,
‘CL’	Castilla-León,
‘CM’	Castilla-La-Mancha,
‘CV’	Valencia,
‘EXT’	Extremadura,
‘G’	Galicia,
‘IB’	Baleares,
‘IC’	Canarias,
‘LR’	La Rioja,
‘M’	Madrid,
‘MU’	Murcia,
‘NA’	Navarra,
‘PV’	País Vasco

Generation 2:

The RegionAlpha codes shall be held on a list maintained on the website of the laboratory appointed to carry out interoperability testing.

2.122. RegionNumeric

Numerical reference to a region within a specified country.

RegionNumeric ::= OCTET STRING (SIZE(1))

Generation 1:

Value assignment:

‘00’H No information available,

Spain:

‘01’H Andalucía,

'02'H	Aragón,
'03'H	Asturias,
'04'H	Cantabria,
'05'H	Cataluña,
'06'H	Castilla-León,
'07'H	Castilla-La-Mancha,
'08'H	Valencia,
'09'H	Extremadura,
'0A'H	Galicia,
'0B'H	Baleares,
'0C'H	Canarias,
'0D'H	La Rioja,
'0E'H	Madrid,
'0F'H	Murcia,
'10'H	Navarra,
'11'H	País Vasco

Generation 2:

The Region-Numeric codes shall be held on a list maintained on the website of the laboratory appointed to carry out interoperability testing.

2.123. RemoteCommunicationModuleSerialNumber

Generation 2:

Serial number of the Remote Communication Module.

RemoteCommunicationModuleSerialNumber ::= ExtendedSerialNumber

2.124. RSAKeyModulus

Generation 1:

The modulus of a RSA key pair.

RSAKeyModulus ::= OCTET STRING (SIZE(128))

Value assignment: Unspecified.

2.125. RSAKeyPrivateExponent

Generation 1:

The private exponent of a RSA key pair.

RSAKeyPrivateExponent ::= OCTET STRING (SIZE(128))

Value assignment: Unspecified.

2.126. RSAKeyPublicExponent

Generation 1:

The public exponent of a RSA key pair.

RSAKeyPublicExponent ::= OCTET STRING (SIZE(8))

Value assignment: Unspecified.

2.127. RtmData

Generation 2:

For the definition of this data type see ~~Appendix~~ **Sub-appendix** 14.

2.128. SealDataCard

Generation 2:

This data type stores information about the seals that are attached to the different components of a vehicle and is intended for storage on a card. This data type is related to ~~Annex~~ **Appendix** 1C requirement 337.

SealDataCard ::= SEQUENCE {

noOfSealRecords INTEGER(1..5),

sealRecords SET SIZE(noOfSealRecords) OF SealRecord

}

noOfSealRecords is the number of records in sealRecords.

sealRecords is a set of seal records.

2.129. SealDataVu

Generation 2:

This data type stores information about the seals that are attached to the different components of a vehicle and is intended for storage in a Vehicle Unit.

SealDataVu ::= SEQUENCE SIZE(5) OF {

sealRecords SealRecord

}

sealRecords is a set of seal records. If there are less than 5 seals available the value of the EquipmentType in all unused sealRecords shall be set to 156, i.e. unused.

2.130. SealRecord

Generation 2:

This data type stores information about a seal that is attached to a component. This data type is related to ~~Annex~~ **Appendix** 1C requirement 337.

SealRecord ::= SEQUENCE {

```

equipmentType          EquipmentType,
extendedSealIdentifier ExtendedSealIdentifier
}

```

equipmentType identifies the type of equipment the seal is attached to.

extendedSealIdentifier is the identifier of the seal attached to the equipment.

2.131. SensorApprovalNumber

Type approval number of the sensor.

Generation 1:

SensorApprovalNumber ::= IA5String(SIZE(8))

Value assignment: Unspecified.

Generation 2:

SensorApprovalNumber ::= IA5String(SIZE(16))

Value assignment:

The approval number shall be provided as published on the corresponding ~~European Commission web-site~~ **i.e. website run by the laboratory competent for interoperability tests, i.e.** for example including hyphens if any. The approval number shall be left-aligned.

2.132. SensorExternalGNSSApprovalNumber

Generation 2:

Type approval number of the external GNSS facility.

SensorExternalGNSSApprovalNumber ::= IA5String(SIZE(16))

Value assignment:

The approval number shall be provided as published on the corresponding ~~European Commission web-site~~ **run by the laboratory competent for interoperability tests**, i.e. for example including hyphens if any. The approval number shall be left-aligned.

2.133. SensorExternalGNSSCoupledRecord

Generation 2:

Information, stored in a vehicle unit, related to the identification of the external GNSS facility coupled with the vehicle unit (~~Annex~~ **Appendix 1C** requirement 100).

SensorExternalGNSSCoupledRecord ::= SEQUENCE {

sensorSerialNumber SensorGNSSSerialNumber,

sensorApprovalNumber SensorExternalGNSSApprovalNumber,

sensorCouplingDate SensorGNSSCouplingDate

}

sensorSerialNumber is the serial number of the external GNSS facility coupled with the vehicle unit.

sensorApprovalNumber is the approval number of this external GNSS facility.

sensorCouplingDate is a date of coupling of this external GNSS facility with the vehicle unit.

2.134. SensorExternalGNSSIdentification

Generation 2:

Information related to the identification of the external GNSS facility (~~Annex~~ **Appendix 1C** requirement 98).

```
SensorExternalGNSSIdentification ::= SEQUENCE {
    sensorSerialNumber          SensorGNSSSerialNumber,
    sensorApprovalNumber       SensorExternalGNSSApprovalNumber,
    sensorSCIdentifier          SensorExternalGNSSSCIdentifier,
    sensorOSIdentifier         SensorExternalGNSSOSIdentifier
}
```

sensorSerialNumber is the extended serial number of the external GNSS facility.

sensorApprovalNumber is the approval number of the external GNSS facility.

sensorSCIdentifier is the identifier of the security component of the external GNSS facility.

sensorOSIdentifier is the identifier of the operating system of the external GNSS facility.

2.135. SensorExternalGNSSInstallation

Generation 2:

Information, stored in an external GNSS facility, related to the installation of the external GNSS sensor (~~Annex~~ **Appendix 1C** requirement 123).

```
SensorExternalGNSSInstallation ::= SEQUENCE {
    sensorCouplingDateFirst    SensorGNSSCouplingDate,
    firstVuApprovalNumber     VuApprovalNumber,
    firstVuSerialNumber       VuSerialNumber,
    sensorCouplingDateCurrent  SensorGNSSCouplingDate,
    currentVuApprovalNumber   VuApprovalNumber,
    currentVUSerialNumber     VuSerialNumber
}
```

sensorCouplingDateFirst is the date of the first coupling of external GNSS facility with a vehicle unit.

firstVuApprovalNumber is the approval number of the first vehicle unit coupled with the external GNSS facility.

firstVuSerialNumber is the serial number of the first vehicle unit paired with the external GNSS facility.

sensorCouplingDateCurrent is the date of the current coupling of external GNSS facility with a vehicle unit.

currentVuApprovalNumber is the approval number of the vehicle unit currently coupled with the external GNSS facility.

currentVUSerialNumber is the serial number of the vehicle unit currently coupled with the external GNSS facility.

2.136. **SensorExternalGNSSOSIdentifier**

Generation 2:

Identifier of the operating system of the external GNSS facility.

SensorOSIdentifier ::= IA5String(SIZE(2))

Value assignment: manufacturer specific.

2.137. **SensorExternalGNSSSCIIdentifier**

Generation 2:

This type is used e.g. to identify the cryptographic module of the external GNSS facility.

Identifier of the security component of the external GNSS facility.

SensorExternalGNSSSCIIdentifier ::= IA5String(SIZE(8))

Value assignment: component manufacturer specific.

2.138. **SensorGNSSCouplingDate**

Generation 2:

Date of a coupling of the external GNSS facility with a vehicle unit.

SensorGNSSCouplingDate ::= TimeReal

Value assignment: Unspecified.

2.139. **SensorGNSSSerialNumber**

Generation 2:

This type is used to store the serial number of the GNSS receiver both when it is inside the VU and when it is outside the VU.

Serial number of the GNSS receiver.

SensorGNSSSerialNumber ::= ExtendedSerialNumber

2.140. **SensorIdentification**

Information, stored in a motion sensor, related to the identification of the motion sensor (~~Annex~~ **Appendix** 1B requirement 077 and ~~Annex~~ **Appendix** 1C requirement 95).

SensorIdentification ::= SEQUENCE {

sensorSerialNumber	SensorSerialNumber,
sensorApprovalNumber	SensorApprovalNumber,
sensorSCIdentifier	SensorSCIdentifier,
sensorOSIdentifier	SensorOSIdentifier
}	

sensorSerialNumber is the extended serial number of the motion sensor (includes part number and manufacturer code).

sensorApprovalNumber is the approval number of the motion sensor.

sensorSCIdentifier is the identifier of the security component of the motion sensor.

sensorOSIdentifier is the identifier of the operating system of the motion sensor.

2.141. SensorInstallation

Information, stored in a motion sensor, related to the installation of the motion sensor (~~Annex~~ **Appendix 1B** requirement 099 and ~~Annex~~ **Appendix 1C** requirement 122).

```
SensorInstallation ::= SEQUENCE {
  sensorPairingDateFirst      SensorPairingDate,
  firstVuApprovalNumber      VuApprovalNumber,
  firstVuSerialNumber        VuSerialNumber,
  sensorPairingDateCurrent   SensorPairingDate,
  currentVuApprovalNumber    VuApprovalNumber,
  currentVUSerialNumber      VuSerialNumber
}
```

sensorPairingDateFirst is the date of the first pairing of the motion sensor with a vehicle unit.

firstVuApprovalNumber is the approval number of the first vehicle unit paired with the motion sensor.

firstVuSerialNumber is the serial number of the first vehicle unit paired with the motion sensor.

sensorPairingDateCurrent is the date of the current pairing of the motion sensor with the vehicle unit.

currentVuApprovalNumber is the approval number of the vehicle unit currently paired with the motion sensor.

currentVUSerialNumber is the serial number of the vehicle unit currently paired with the motion sensor.

2.142. SensorInstallationSecData

Information, stored in a workshop card, related to the security data needed for pairing motion sensors to vehicle units (~~Annex~~ **Appendix 1C** requirement 308 and 331).

Generation 1:

SensorInstallationSecData ::= TDesSessionKey

Value assignment: in accordance with ISO 16844-3.

Generation 2:

As described in ~~Appendix~~ **Sub-appendix** 11 a workshop card shall store up to three keys for VU Motion Sensor pairing. These keys have different key versions.

```
SensorInstallationSecData ::= SEQUENCE {
  kWCKKey1           KMWCKKey,
  kWCKKey2           KMWCKKey OPTIONAL,
  kWCKKey3           KMWCKKey OPTIONAL
}
```

2.143. SensorOSIdentifier

Identifier of the operating system of the motion sensor.

SensorOSIdentifier ::= IA5String(SIZE(2))

Value assignment: manufacturer specific.

2.144. SensorPaired

Generation 1:

Information, stored in a vehicle unit, related to the identification of the motion sensor paired with the vehicle unit (~~Annex~~ **Appendix** 1B requirement 079).

```
SensorPaired ::= SEQUENCE {
  sensorSerialNumber      SensorSerialNumber,
  sensorApprovalNumber    SensorApprovalNumber,
  sensorPairingDateFirst  SensorPairingDate
}
```

sensorSerialNumber is the serial number of the motion sensor currently paired with the vehicle unit.

sensorApprovalNumber is the approval number of the motion sensor currently paired with the vehicle unit.

sensorPairingDateFirst is the date of the first pairing with a vehicle unit of the motion sensor currently paired with the vehicle unit.

2.145. SensorPairedRecord

Generation 2:

Information, stored in a vehicle unit, related to the identification of a motion sensor paired with the vehicle unit (~~Annex~~ **Appendix** 1C requirement 97).

```
SensorPairedRecord ::= SEQUENCE {
  sensorSerialNumber      SensorSerialNumber,
```

```
sensorApprovalNumber      SensorApprovalNumber,  
sensorPairingDate         SensorPairingDate  
}
```

sensorSerialNumber is the serial number of a motion sensor paired with the vehicle unit.

sensorApprovalNumber is the approval number of this motion sensor.

sensorPairingDate is a date of pairing of this motion sensor with the vehicle unit.

2.146. SensorPairingDate

Date of a pairing of the motion sensor with a vehicle unit.

SensorPairingDate ::= TimeReal

Value assignment: Unspecified.

2.147. SensorSCIdentifier

Identifier of the security component of the motion sensor.

SensorSCIdentifier ::= IA5String(SIZE(8))

Value assignment: component manufacturer specific.

2.148. SensorSerialNumber

Serial number of the motion sensor.

SensorSerialNumber ::= ExtendedSerialNumber

2.149. Signature

A digital signature.

Generation 1:

Signature ::= OCTET STRING (SIZE(128))

Value assignment: in accordance with ~~Appendix~~ **Sub-appendix** 11 Common security mechanisms.

Generation 2:

Signature ::= OCTET STRING (SIZE(64..132))

Value assignment: in accordance with ~~Appendix~~ **Sub-appendix** 11 Common security mechanisms.

2.150. SignatureRecordArray

Generation 2:

A set of signatures plus metadata used in the download protocol.

SignatureRecordArray ::= SEQUENCE {

```

recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records            SET SIZE(noOfRecords) OF Signature
}

```

recordType denotes the type of the record (Signature). **Value Assignment:** See RecordType
recordSize is the size of the Signature in bytes.

noOfRecords is the number of records in the set records. The value shall be set to 1 as the signatures may have different lengths.

records is the set of signatures.

2.151. SimilarEventsNumber

The number of similar events for one given day (~~Annex Appendix~~ 1B requirement 094 and ~~Annex Appendix~~ 1C requirement 117).

```
SimilarEventsNumber ::= INTEGER(0..255)
```

Value assignment: 0 is not used, 1 means that only one event of that type has occurred and has been stored on that day, 2 means that 2 events of that type has occurred on that day (one only has been stored), ...255 means that 255 or more events of that type have occurred on that day.

2.152. SpecificConditionRecord

Information, stored in a driver card, a workshop card or a vehicle unit, related to a specific condition (requirements ~~Annex Appendix~~ 1C 130, 276, 301, 328, and 355).

```

SpecificConditionRecord ::= SEQUENCE {
entryTime          TimeReal,
specificConditionType SpecificConditionType
}

```

entryTime is the date and time of the entry.

specificConditionType is the code identifying the specific condition.

2.153. SpecificConditions

Information, stored in a driver card, a workshop card or a vehicle unit, related to a specific condition (~~Annex Appendix~~ 1C requirement 131, 277, 302, 329, and 356).

Generation 2:

```

SpecificConditions := SEQUENCE {
conditionPointerNewestRecord
                                INTEGER(0..NoOfSpecificConditionRecords-1),
specificConditionRecords      SET   SIZE(NoOfSpecificConditionRecords)   OF
SpecificConditionRecord
}

```

}

conditionPointerNewestRecord is the index of the last updated specific condition record.

Value assignment: Number corresponding to the numerator of the specific condition record, beginning with '0' for the first occurrence of the specific condition record in the structure.

specificConditionRecords is the set of records containing information on the specific conditions recorded.

2.154. SpecificConditionType

Code identifying a specific condition (~~Annex Appendix 1B requirements 050b, 105a, 212a and 230a and Annex Appendix 1C requirements-requirement 62~~).

SpecificConditionType ::= INTEGER(0..255)

Generation 1:

Value assignment:

'00'H	RFU
'01'H	Out of scope – Begin
'02'H	Out of scope – End
'03'H	Ferry / Train crossing
'04'H .. 'FF'H	RFU

Generation 2:

Value assignment:

'00'H	RFU
'01'H	Out of scope – Begin
'02'H	Out of scope – End
'03'H	Ferry / Train crossing – Begin
'04'H	Ferry / Train crossing – End
'05'H 'FF'H	RFU

2.155. Speed

Speed of the vehicle (km/h).

Speed ::= INTEGER(0..255)

Value assignment: kilometers per hour in the operational range 0 to 220 km/h.

2.156. SpeedAuthorised

Maximum authorised Speed of the vehicle (definition hh)).

SpeedAuthorised ::= Speed

2.157. SpeedAverage

Average speed in a previously defined duration (km/h).

SpeedAverage ::= Speed

2.158. SpeedMax

Maximum speed measured in a previously defined duration.

SpeedMax ::= Speed

2.158a. TachographCardsGen1Suppression

Generation 2, version 2:

Ability of a second generation VU to use first generation of driver, control and company cards (see Sub-appendix 15, MIG_002).

TachographCardsGen1Suppression ::= INTEGER (0..2¹⁶-1)

Value assignment:

‘0000’H The VU is able to use the generation 1 of tachograph cards (default value),

‘A5E3’H The VU is not able to use the tachograph cards generation 1,

All other values Not used.

2.159. TachographPayload

Generation 2:

For the definition of this data type see Appendix Sub-appendix 14.

2.160. Reserved for future use

TachographPayloadEncrypted

Generation 2

The DER TLV encrypted tachograph payload i.e. the data sent encrypted in the RTM message. For the encryption see Appendix 11 Part B chapter 13

TachographPayloadEncrypted SEQUENCE

OCTET STRING (SIZE)(1)

Length OCTET STRING (SIZE 1.2.160 Reserved for future use 2)

paddingContentIndicatorByte OCTET STRING (SIZE (1))

encryptedData OCTET STRING (SIZE)(16.192)

tag is part of the DER TLV encoding and shall be set to 87 (see Appendix 11 Part B chapter 13)

length is part of the DER TLV encoding and shall encode the length of the following padding

ContentIndicated Byte and the encryptedData

paddingContentIndicatorByte shall be set to 00

~~encryptedData is encrypted tachographPayload as specified in Appendix 11 Part B chapter 13. The length of the data in octet shall always be a multiple of 16~~

2.161. TDesSessionKey

Generation 1:

A triple DES session key.

```
TDesSessionKey ::= SEQUENCE {
tDesKeyA      OCTET STRING (SIZE(8)),
tDesKeyB      OCTET STRING (SIZE(8))
}
```

Value assignment: not further specified.

2.162. TimeReal

Code for a combined date and time field, where the date and time are expressed as seconds past 00h.00m.00s. on 1 January 1970 ~~GMT~~UTC.

```
TimeReal{INTEGER:TimeRealRange} ::= INTEGER (0..TimeRealRange)
```

Value assignment – Octet Aligned: Number of seconds since midnight 1 January 1970 ~~GMT~~UTC.

The max. possible date/time is in the year 2106.

2.163. TyreSize

Designation of tyre dimensions.

```
TyreSize ::= IA5String(SIZE(15))
```

Value assignment: in accordance with ~~Directive 93/23 (ECE) 31/03/92 O. J. L129 p.95~~ **UNECE Regulation 54**

2.164. VehicleIdentificationNumber

Vehicle Identification Number (VIN) referring to the vehicle as a whole, normally chassis serial number or frame number.

```
VehicleIdentificationNumber ::= IA5String(SIZE(17))
```

Value assignment: As defined in ISO 3779.

2.165. VehicleIdentificationNumberRecordArray

Generation 2:

The Vehicle Identification Number plus metadata as used in the download protocol.

```
VehicleIdentificationNumberRecordArray ::= SEQUENCE {
```



```

recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF
VehicleIdentificationNumber
}

```

recordType denotes the type of the record (VehicleIdentificationNumber). **Value Assignment:** See RecordType

recordSize is the size of the VehicleIdentificationNumber in bytes.

noOfRecords is the number of records in the set records.

records is the set of vehicle identification numbers.

2.166. VehicleRegistrationIdentification

Identification of a vehicle, unique for Europe (VRN and ~~Member State~~ **Contracting Party**).

```

VehicleRegistrationIdentification ::= SEQUENCE {
vehicleRegistrationNation      NationNumeric,
vehicleRegistrationNumber      VehicleRegistrationNumber
}

```

vehicleRegistrationNation is the nation where the vehicle is registered.

vehicleRegistrationNumber is the registration number of the vehicle (VRN).

2.166a. VehicleRegistrationIdentificationRecordArray

Generation 2, version 2:

The Vehicle Registration Identification plus metadata as used in the download protocol.

```

VehicleRegistrationIdentificationRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records             SET          SIZE(noOfRecords)          OF
VehicleRegistrationIdentification
}

```

recordType denotes the type of the record (VehicleRegistrationIdentification). **Value assignment:** see RecordType.

recordSize is the size of the VehicleRegistrationIdentification in bytes.

noOfRecords is the number of records in the set records.

records is the set of vehicle registration identification.

2.167. VehicleRegistrationNumber

Registration number of the vehicle (VRN). The registration number is assigned by the vehicle licensing authority.

```
VehicleRegistrationNumber ::= SEQUENCE {
codePage    INTEGER (0..255),
vehicleRegNumber OCTET STRING (SIZE(13))
}
```

codePage specifies a character set defined in Chapter 4,

vehicleRegNumber is a VRN encoded using the specified character set.

Value assignment: Country specific.

2.168. VehicleRegistrationNumberRecordArray

Generation 2, **version 1:**

The Vehicle Registration Number plus metadata as used in the download protocol.

```
VehicleRegistrationNumberRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF
VehicleRegistrationNumber
}
```

recordType denotes the type of the record (VehicleRegistrationNumber). **Value Assignment:** See RecordType

recordSize is the size of the VehicleRegistrationNumber in bytes.

noOfRecords is the number of records in the set records.

records is the set of vehicle registration numbers.

2.169. VuAbility

Generation 2:

Information stored in a VU on the ability of the VU to use generation 1 tachograph cards or not (~~Annex~~ **Appendix 1C** requirement 121).

```
VuAbility ::= OCTET STRING (SIZE(1))
```

Value assignment – Octet Aligned : ‘xxxxxxa’B (8 bits)

For the ability to support of generation 1:

```
‘a’B      Ability to support generation 1 tachograph cards:
‘0’ B     Generation 1 is supported,
‘1’ B     Generation1 is not supported,
```

'xxxxxxx'B RFU

2.170. VuActivityDailyData

Generation 1:

Information, stored in a VU, related to changes of activity and/or changes of driving status and/or changes of card status for a given calendar day (~~Annex~~ **Appendix 1B** requirement 084 and ~~Annex~~ **Appendix 1C** requirement 105, 106, 107) and to slots status at 00:00 that day.

```
VuActivityDailyData ::= SEQUENCE {
noOfActivityChanges          INTEGER SIZE(0..1440),
activityChangeInfos          SET          SIZE(noOfActivityChanges)          OF
ActivityChangeInfo
}
```

noOfActivityChanges is the number of ActivityChangeInfo words in the activityChangeInfos set.

activityChangeInfos is the set of ActivityChangeInfo words stored in the VU for the day. It always includes two ActivityChangeInfo words giving the status of the two slots at 00:00 that day.

2.171. VuActivityDailyRecordArray

Generation 2:

Information, stored in a VU, related to changes of activity and/or changes of driving status and/or changes of card status for a given calendar day (~~Annex~~ **Appendix 1C** requirement 105, 106, 107) and to slots status at 00:00 that day.

```
VuActivityDailyRecordArray ::= SEQUENCE {
recordType                   RecordType,
recordSize                   INTEGER(1..65535),
noOfRecords                  INTEGER(0..65535),
records                      SET SIZE(noOfRecords) OF ActivityChangeInfo
}
```

recordType denotes the type of the record (ActivityChangeInfo). **Value Assignment:** See RecordType

recordSize is the size of the ActivityChangeInfo in bytes.

noOfRecords is the number of records in the set records.

records is the set of ActivityChangeInfo words stored in the VU for the day. It always includes two ActivityChangeInfo words giving the status of the two slots at 00:00 that day.

2.172. VuApprovalNumber

Type approval number of the vehicle unit.

Generation 1:

VuApprovalNumber ::= IA5String(SIZE(8))

Value assignment: Unspecified.

Generation 2:

VuApprovalNumber ::= IA5String(SIZE(16))

Value assignment:

The approval number shall be provided as published on the ~~corresponding European Commission web site, i.e. website of the laboratory appointed to carry out interoperability testing, i.e.~~ for example including hyphens if any. The approval number shall be left-aligned.

2.173. VuCalibrationData

Generation 1:

Information, stored in a vehicle unit, related to the calibrations of the ~~recording equipment control device~~ (~~Annex Appendix~~ 1B requirement 098).

VuCalibrationData ::= SEQUENCE {

noOfVuCalibrationRecords INTEGER(0..255),

vuCalibrationRecords SET SIZE(noOfVuCalibrationRecords) OF
VuCalibrationRecord

}

noOfVuCalibrationRecords is the number of records contained in the vuCalibrationRecords set.

vuCalibrationRecords is the set of calibration records.

2.174. VuCalibrationRecord

Information, stored in a vehicle unit, related a calibration of the ~~recording equipment control device~~ (~~Annex Appendix~~ 1B requirement 098 and ~~Annex Appendix~~ 1C requirement 119 and 120).

Generation 1:

VuCalibrationRecord ::= SEQUENCE {

calibrationPurpose CalibrationPurpose,

workshopName Name,

workshopAddress Address,

workshopCardNumber FullCardNumber,

workshopCardExpiryDate TimeReal,

vehicleIdentificationNumber VehicleIdentificationNumber,

vehicleRegistrationIdentification VehicleRegistrationIdentification,

wVehicleCharacteristicConstant W-VehicleCharacteristicConstant,

kConstantOfRecordingEquipment K-ConstantOfRecordingEquipment,

lTyreCircumference	L-TyreCircumference,
tyreSize	TyreSize,
authorisedSpeed	SpeedAuthorised,
oldOdometerValue	OdometerShort,
newOdometerValue	OdometerShort,
oldTimeValue	TimeReal,
newTimeValue	TimeReal,
nextCalibrationDate	TimeReal

}

calibrationPurpose is the purpose of the calibration.

workshopName, **workshopAddress** are the workshop name and address.

workshopCardNumber identifies the workshop card used during the calibration.

workshopCardExpiryDate is the card expiry date.

vehicleIdentificationNumber is the VIN.

vehicleRegistrationIdentification contains the VRN and registering ~~Member State~~ **Contracting Party**.

wVehicleCharacteristicConstant is the characteristic coefficient of the vehicle.

kConstantOfRecordingEquipment is the constant of the ~~recording equipment~~ **control device**.

lTyreCircumference is the effective circumference of the wheel tyres.

tyreSize is the designation of the dimension of the tyres mounted on the vehicle

authorisedSpeed is the authorised speed of the vehicle.

oldOdometerValue, **newOdometerValue** are the old and new values of the odometer.

oldTimeValue, **newTimeValue** are the old and new values of date and time.

nextCalibrationDate is the date of the next calibration of the type specified in **CalibrationPurpose** to be carried out by the authorised inspection authority.

Generation 2, **version 1**:

```
VuCalibrationRecord ::= SEQUENCE {
  calibrationPurpose          CalibrationPurpose,
  workshopName                Name,
  workshopAddress             Address,
  workshopCardNumber          FullCardNumber,
  workshopCardExpiryDate      TimeReal,
  vehicleIdentificationNumber VehicleIdentificationNumber,
  vehicleRegistrationIdentification VehicleRegistrationIdentification,
  wVehicleCharacteristicConstant W-VehicleCharacteristicConstant,
  kConstantOfRecordingEquipment K-ConstantOfRecordingEquipment,
```

lTyreCircumference	L-TyreCircumference,
tyreSize	TyreSize,
authorisedSpeed	SpeedAuthorised,
oldOdometerValue	OdometerShort,
newOdometerValue	OdometerShort,
oldTimeValue	TimeReal,
newTimeValue	TimeReal,
nextCalibrationDate	TimeReal,
sealDataVu	SealDataVu
}	

In addition to generation 1 the following data element is used:

sealDataVu gives information about the seals that are attached to different components of the vehicle.

Generation 2, version 2:

VuCalibrationRecord ::= SEQUENCE {	
calibrationPurpose	CalibrationPurpose,
workshopName	Name,
workshopAddress	Address,
workshopCardNumber	FullCardNumber,
workshopCardExpiryDate	TimeReal,
vehicleIdentificationNumber	VehicleIdentificationNumber,
vehicleRegistrationIdentification	VehicleRegistrationIdentification,
wVehicleCharacteristicConstant	W-VehicleCharacteristicConstant,
kConstantOfRecordingEquipment	K-ConstantOfRecordingEquipment,
lTyreCircumference	L-TyreCircumference,
tyreSize	TyreSize,
authorisedSpeed	SpeedAuthorised,
oldOdometerValue	OdometerShort,
newOdometerValue	OdometerShort,
oldTimeValue	TimeReal,
newTimeValue	TimeReal,
nextCalibrationDate	TimeReal,
sensorSerialNumber	SensorSerialNumber,
sensorGNSSSerialNumber	SensorGNSSSerialNumber,
rcmSerialNumber	RemoteCommunicationModuleSerialNumber,
sealDataVu	SealDataVu,

byDefaultLoadType	LoadType,
calibrationCountry	NationNumeric,
calibrationCountryTimestamp	TimeReal

}

In addition to generation 1 the following data element is used:

sensorSerialNumber is the serial number of the motion sensor paired with the vehicle unit at the end of the calibration,

sensorGNSSSerialNumber is the serial number of the external GNSS facility coupled with the vehicle unit at the end of the calibration (if any),

rcmSerialNumber is the serial number of the remote communication facility coupled with the vehicle unit at the end of the calibration (if any),

sealDataVu gives information about the seals that are attached to different components of the vehicle.

byDefaultLoadType is the by-default load type of the vehicle (only present in version 2).

calibrationCountry is the country in which the calibration has been performed.

calibrationCountryTimestamp is the date and time when the position used to determine the country in which the calibration has been performed was provided by the GNSS receiver.

2.175. VuCalibrationRecordArray

Generation 2:

Information, stored in a vehicle unit, related to the calibrations of the ~~recording equipment~~ **control device** (Annex Appendix 1C requirement 119 and 120).

```
VuCalibrationRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords         INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF VuCalibrationRecord
}
```

recordType denotes the type of the record (VuCalibrationRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuCalibrationRecord in bytes.

noOfRecords is the number of records in the set records.

records is the set of calibration records.

2.176. VuCardIWData

Generation 1:

Information, stored in a vehicle unit, related to insertion and withdrawal cycles of driver cards or of workshop cards in the vehicle unit (~~Annex~~ **Appendix** 1B requirement 081 and ~~Annex~~ **Appendix** 1C requirement 103).

```
VuCardIWData ::= SEQUENCE {
noOfIWRecords          INTEGER(0..216-1),
vuCardIWRecords        SET SIZE(noOfIWRecords) OF VuCardIWRecord
}
```

noOfIWRecords is the number of records in the set vuCardIWRecords.

vuCardIWRecords is a set of records related to card insertion withdrawal cycles.

2.177. VuCardIWRecord

Information, stored in a vehicle unit, related to an insertion and withdrawal cycle of a driver card or of a workshop card in the vehicle unit (~~Annex~~ **Appendix** 1B requirement 081 and ~~Annex~~ **Appendix** 1C requirement 102).

Generation 1:

```
VuCardIWRecord ::= SEQUENCE {
cardHolderName          HolderName,
fullCardNumber          FullCardNumber,
cardExpiryDate          TimeReal,
cardInsertionTime       TimeReal,
vehicleOdometerValueAtInsertion OdometerShort,
cardSlotNumber          CardSlotNumber,
cardWithdrawalTime      TimeReal,
vehicleOdometerValueAtWithdrawal OdometerShort,
previousVehicleInfo     PreviousVehicleInfo,
manualInputFlag         ManualInputFlag
}
```

cardHolderName is the driver or workshop card holder's surname and first names as stored in the card.

fullCardNumber is the type of card, its issuing ~~Member State~~ **Contracting Party** and its card number as stored in the card.

cardExpiryDate is the card's expiry date as stored in the card.

cardInsertionTime is the insertion date and time.

vehicleOdometerValueAtInsertion is the vehicle odometer value at card insertion.

cardSlotNumber is the slot in which the card is inserted.

cardWithdrawalTime is the withdrawal date and time.

vehicleOdometerValueAtWithdrawal is the vehicle odometer value at card withdrawal.

previousVehicleInfo contains information about the previous vehicle used by the driver, as stored in the card.

manualInputFlag is a flag identifying if the cardholder has manually entered driver activities at card insertion.

Generation 2:

```
VuCardIWRecord ::= SEQUENCE {
cardHolderName           HolderName,
fullCardNumberAndGeneration FullCardNumberAndGeneration,
cardExpiryDate           TimeReal,
cardInsertionTime        TimeReal,
vehicleOdometerValueAtInsertion OdometerShort,
cardSlotNumber           CardSlotNumber,
cardWithdrawalTime       TimeReal,
vehicleOdometerValueAtWithdrawal OdometerShort,
previousVehicleInfo      PreviousVehicleInfo,
manualInputFlag          ManualInputFlag
}
```

Instead of fullCardNumber the generation 2 data structure makes use of the following data element.

fullCardNumberAndGeneration is the type of card, its issuing ~~Member State~~ **Contracting Party**, its card number and generation as stored in the card.

2.178. VuCardIWRecordArray

Generation 2:

Information, stored in a vehicle unit, related to insertion and withdrawal cycles of driver cards or of workshop cards in the vehicle unit (~~Annex~~ **Appendix 1C** requirement 103).

```
VuCardIWRecordArray ::= SEQUENCE {
recordType               RecordType,
recordSize               INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                  SET SIZE(noOfRecords) OF VuCardIWRecord
}
```

recordType denotes the type of the record (VuCardIWRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuCardIWRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of records related to card insertion withdrawal cycles.

2.179. VuCardRecord

Generation 2:

Information, stored in a vehicle unit, about a tachograph card used (~~Annex~~ **Appendix 1C** requirement 132).

VuCardRecord ::= SEQUENCE {

cardNumberAndGenerationInformation	FullCardNumberAndGeneration,
cardExtendedSerialNumber	ExtendedSerialNumber,
cardPersonaliserID	OCTET STRING(SIZE(1)),
typeOfTachographCardID	EquipmentType,
cardStructureVersion	CardStructureVersion,
cardNumber	CardNumber

}

cardNumberAndGenerationInformation is the full card number and generation of the card used (data type 2.74).

cardExtendedSerialNumber as read from the file EF_ICC under the MF of the card.

~~**cardPersonaliserID** as read from the file EF_ICC under the MF of the card.~~

~~**typeOfTachographCardID** as read from the file EF_Application_Identification under the DF_Tachograph_G2~~

cardStructureVersion as read from the file EF_Application_Identification under the DF_Tachograph_G2.

cardNumber as read from the file EF_Identification under the DF_Tachograph_G2.

2.180. VuCardRecordArray

Generation 2:

Information stored in a vehicle unit about the tachograph cards used with this VU. This information is intended for the analysis of VU – card problems (~~Annex~~ **Appendix 1C** requirement 132).

VuCardRecordArray ::= SEQUENCE {

recordType	RecordType,
recordSize	INTEGER(1..65535),
noOfRecords	INTEGER(0..65535),
records	SET SIZE(noOfRecords) OF VuCardRecord

}

recordType denotes the type of the record (VuCardRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuCardRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of records related to the tachograph cards used with the VU.

2.181. VuCertificate

Certificate of the public key of a vehicle unit.

VuCertificate ::= Certificate

2.182. VuCertificateRecordArray

Generation 2:

The VU certificate plus metadata as used in the download protocol.

```
VuCertificateRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords        INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF VuCertificate
}
```

recordType denotes the type of the record (VuCertificate). **Value Assignment:** See RecordType

recordSize is the size of the VuCertificate in bytes.

noOfRecords is the number of records in the set records. The value shall be set to 1 as the certificates may have different lengths.

records is a set of VU certificates.

2.183. VuCompanyLocksData

Generation 1:

Information, stored in a vehicle unit, related to company locks (~~Annex~~ **Appendix 1B** requirement 104).

```
VuCompanyLocksData ::= SEQUENCE {
noOfLocks           INTEGER(0..255),
vuCompanyLocksRecords SET SIZE(noOfLocks) OF VuCompanyLocksRecord
}
```

noOfLocks is the number of locks listed in vuCompanyLocksRecords.

vuCompanyLocksRecords is the set of company locks records.

2.184. VuCompanyLocksRecord

Information, stored in a vehicle unit, related to one company lock (~~Annex~~ **Appendix 1B** requirement 104 and ~~Annex~~ **Appendix 1C** requirement 128).

Generation 1:

```
VuCompanyLocksRecord ::= SEQUENCE {
```

```

lockInTime           TimeReal,
lockOutTime          TimeReal,
companyName          Name
companyAddress        Address,
companyCardNumber    FullCardNumber
}

```

lockInTime, **lockOutTime** are the date and time of lock-in and lock-out.

companyName, **companyAddress** are the company name and address related with the lock-in.

companyCardNumber identifies the card used at lock-in.

Generation 2:

```

VuCompanyLocksRecord ::= SEQUENCE {
lockInTime           TimeReal,
lockOutTime          TimeReal,
companyName          Name,
companyAddress        Address,
companyCardNumberAndGeneration FullCardNumberAndGeneration
}

```

Instead of **companyCardNumber** the generation 2 data structure makes use of the following data element.

companyCardNumberAndGeneration identifies the card including its generation used at lock-in.

2.185. VuCompanyLocksRecordArray

Generation 2:

Information, stored in a vehicle unit, related to company locks (~~Annex~~ **Appendix 1C** requirement 128).

```

VuCompanyLocksRecordArray ::= SEQUENCE {
recordType           RecordType,
recordSize           INTEGER(1..65535),
noOfRecords          INTEGER(0..65535),
records              SET          SIZE(noOfRecords)      OF
VuCompanyLocksRecord
}

```

recordType denotes the type of the record (VuCompanyLocksRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuCompanyLocksRecord in bytes.

noOfRecords is the number of records in the set records. Value 0..255.

records is the set of company locks records.

2.185a. VuConfigurationLengthRange

Generation 2, version 2:

Number of bytes in a tachograph card, available to store VU configurations.

VuConfigurationLengthRange ::= INTEGER(0..2¹⁶-1)

Value assignment: see Sub-appendix 2.

2.186. VuControlActivityData

Generation 1:

Information, stored in a vehicle unit, related to controls performed using this VU (~~Annex~~ **Appendix 1B** requirement 102).

VuControlActivityData ::= SEQUENCE {

noOfControls INTEGER(0..20),

vuControlActivityRecords SET SIZE(noOfControls) OF VuControlActivityRecord

}

noOfControls is the number of controls listed in vuControlActivityRecords.

vuControlActivityRecords is the set of control activity records.

2.187. VuControlActivityRecord

Information, stored in a vehicle unit, related to a control performed using this VU (~~Annex~~ **Appendix 1B** requirement 102 and ~~Annex~~ **Appendix 1C** requirement 126).

Generation 1:

VuControlActivityRecord ::= SEQUENCE {

controlType ControlType,

controlTime TimeReal,

controlCardNumber FullCardNumber,

downloadPeriodBeginTime TimeReal,

downloadPeriodEndTime TimeReal

}

controlType is the type of the control.

controlTime is the date and time of the control.

controlCardNumber identifies the control card used for the control.

downloadPeriodBeginTime is the begin time of the downloaded period, in case of downloading.

downloadPeriodEndTime is the end time of the downloaded period, in case of downloading.

Generation 2:

```
VuControlActivityRecord ::= SEQUENCE {
controlType                ControlType,
controlTime                TimeReal,
controlCardNumberAndGeneration FullCardNumberAndGeneration,
downloadPeriodBeginTime   TimeReal,
downloadPeriodEndTime     TimeReal
}
```

Instead of controlCardNumber the generation 2 data structure makes use of the following data element.

controlCardNumberAndGeneration identifies the control card including its generation used for the control.

2.188. VuControlActivityRecordArray

Generation 2:

Information, stored in a vehicle unit, related to controls performed using this VU (~~Annex~~ **Appendix 1C** requirement 126).

```
VuControlActivityRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET SIZE(noOfRecords) OF VuControlActivityRecord
}
```

recordType denotes the type of the record (VuControlActivityRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuControlActivityRecord in bytes.

noOfRecords is the number of records in the set records.

records is the set of VU control activity records.

2.189. VuDataBlockCounter

Counter, stored in a card, identifying sequentially the insertion withdrawal cycles of the card in vehicle units.

```
VuDataBlockCounter ::= BCDString(SIZE(2))
```

Value assignment: Consecutive Number with max, value 9 999, starting again with 0.

2.190. VuDetailedSpeedBlock

Information, stored in a vehicle unit, related to the vehicle's detailed speed for a minute during which the vehicle has been moving (~~Annex Appendix~~ 1B requirement 093 and ~~Annex Appendix~~ 1C requirement 116).

```
VuDetailedSpeedBlock ::= SEQUENCE {
speedBlockBeginDate      TimeReal,
speedsPerSecond          SEQUENCE SIZE(60) OF Speed
}
```

speedBlockBeginDate is the date and time of the first speed value within the block.

speedsPerSecond is the chronological sequence of measured speeds every seconds for the minute starting at speedBlockBeginDate (included).

2.191. VuDetailedSpeedBlockRecordArray

Generation 2:

Information, stored in a vehicle unit, related to the detailed speed of the vehicle.

```
VuDetailedSpeedBlockRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET SIZE(noOfRecords) OF VuDetailedSpeedBlock
}
```

recordType denotes the type of the record (VuDetailedSpeedBlock). **Value Assignment:** See RecordType

recordSize is the size of the VuDetailedSpeedBlock in bytes.

noOfRecords is the number of records in the set records.

records is the set of detailed speed blocks.

2.192. VuDetailedSpeedData

Generation 1:

Information, stored in a vehicle unit, related to the detailed speed of the vehicle.

```
VuDetailedSpeedData ::= SEQUENCE {
noOfSpeedBlocks          INTEGER(0..216-1),
vuDetailedSpeedBlocks    SET SIZE(noOfSpeedBlocks) OF
                          VuDetailedSpeedBlock
}
```

noOfSpeedBlocks is the number of speed blocks in the vuDetailedSpeedBlocks set.

vuDetailedSpeedBlocks is the set of detailed speed blocks.

2.192a. VuDigitalMapVersion

Generation 2, version 2:

Version of the digital map stored in the vehicle unit (Appendix 1C requirement 133j).

VuDigitalMapVersion ::= IA5String(SIZE(12))

Value assignment: as specified on the dedicated secured website made available by the laboratory competent for interoperability tests (Appendix 1C requirement 133k).

2.193. VuDownloadablePeriod

Oldest and latest dates for which a vehicle unit holds data related to drivers activities (~~Annex~~ **Appendix 1B** requirements 081, 084 or 087 and ~~Annex~~ **Appendix 1C** requirements 102, 105, 108).

VuDownloadablePeriod ::= SEQUENCE {

minDownloadableTime TimeReal

maxDownloadableTime TimeReal

}

minDownloadableTime is the oldest card insertion or activity change or place entry date and time stored in the VU.

maxDownloadableTime is the latest card withdrawal or activity change or place entry date and time stored in the VU.

2.194. VuDownloadablePeriodRecordArray

Generation 2:

The VuDownloadablePeriod plus metadata used in the download protocol.

VuDownloadablePeriodRecordArray ::= SEQUENCE {

recordType RecordType,

recordSize INTEGER(1..65535),

noOfRecords INTEGER(0..65535),

records SET SIZE(noOfRecords) OF VuDownloadablePeriod

}

recordType denotes the type of the record (VuDownloadablePeriod). **Value Assignment:** See RecordType

recordSize is the size of the VuDownloadablePeriod in bytes.

noOfRecords is the number of records in the set records.

records is the set of VuDownloadablePeriod records.

2.195. VuDownloadActivityData

Information, stored in a vehicle unit, related to its last download (~~Annex~~ **Appendix 1B** requirement 105 and ~~Annex~~ **Appendix 1C** requirement 129).

Generation 1:

```
VuDownloadActivityData ::= SEQUENCE {
  downloadingTime      TimeReal,
  fullCardNumber       FullCardNumber,
  companyOrWorkshopName Name
}
```

downloadingTime is the date and time of downloading.

fullCardNumber identifies the card used to authorise the download.

companyOrWorkshopName is the company or workshop name.

Generation 2:

```
VuDownloadActivityData ::= SEQUENCE {
  downloadingTime      TimeReal,
  fullCardNumberAndGeneration FullCardNumberAndGeneration,
  companyOrWorkshopName Name
}
```

Instead of fullCardNumber the generation 2 data structure makes use of the following data element.

fullCardNumberAndGeneration identifies the card including its generation used to authorise the download.

2.196. VuDownloadActivityDataRecordArray

Generation 2:

Information related to the last VU download (~~Annex~~ **Appendix 1C** requirement 129).

```
VuDownloadActivityDataRecordArray ::= SEQUENCE {
  recordType           RecordType,
  recordSize           INTEGER(1..65535),
  noOfRecords          INTEGER(0..65535),
  records              SET SIZE(noOfRecords) OF VuDownloadActivityData
}
```

recordType denotes the type of the record (VuDownloadActivityData). **Value Assignment:** See RecordType

recordSize is the size of the VuDownloadActivityData in bytes.

noOfRecords is the number of records in the set records.

records is the set of download activity data records.

2.197. VuEventData

Generation 1:

Information, stored in a vehicle unit, related to events (~~Annex~~ **Appendix 1B** requirement 094 except over speeding event).

```
VuEventData ::= SEQUENCE {  
noOfVuEvents          INTEGER(0..255),  
vuEventRecords        SET SIZE(noOfVuEvents) OF VuEventRecord  
}
```

noOfVuEvents is the number of events listed in the vuEventRecords set.

vuEventRecords is a set of events records.

2.198. VuEventRecord

Information, stored in a vehicle unit, related to an event (~~Annex~~ **Appendix 1B** requirement 094 and ~~Annex~~ **Appendix 1C** requirement 117 except over speeding event).

Generation 1:

```
VuEventRecord ::= SEQUENCE {  
eventType              EventFaultType,  
eventRecordPurpose     EventFaultRecordPurpose,  
eventBeginTime         TimeReal,  
eventEndTime           TimeReal,  
cardNumberDriverSlotBegin FullCardNumber,  
cardNumberCodriverSlotBegin FullCardNumber,  
cardNumberDriverSlotEnd FullCardNumber,  
cardNumberCodriverSlotEnd FullCardNumber,  
similarEventsNumber    SimilarEventsNumber  
}
```

eventType is the type of the event.

eventRecordPurpose is the purpose for which this event has been recorded.

eventBeginTime is the date and time of beginning of event.

eventEndTime is the date and time of end of event.

cardNumberDriverSlotBegin identifies the card inserted in the driver slot at the beginning of the event.

cardNumberCodriverSlotBegin identifies the card inserted in the co-driver slot at the beginning of the event.

cardNumberDriverSlotEnd identifies the card inserted in the driver slot at the end of the event.

cardNumberCodriverSlotEnd identifies the card inserted in the co-driver slot at the end of the event.

similarEventsNumber is the number of similar events that day.

This sequence can be used for all events other than over speeding events.

Generation 2:

```
VuEventRecord ::= SEQUENCE {
eventType                EventFaultType,
eventRecordPurpose       EventFaultRecordPurpose,
eventBeginTime           TimeReal,
eventEndTime             TimeReal,
cardNumberAndGenDriverSlotBegin FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlotBegin FullCardNumberAndGeneration,
cardNumberAndGenDriverSlotEnd FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlotEnd FullCardNumberAndGeneration,
similarEventsNumber      SimilarEventsNumber,
manufacturerSpecificEventFaultData ManufacturerSpecificEventFaultData
}
```

In addition to generation 1 the following data elements are used:

manufacturerSpecificEventFaultData contains additional, manufacturer specific information about the event.

Instead of **cardNumberDriverSlotBegin**, **cardNumberCodriverSlotBegin**, **cardNumberDriverSlotEnd**, and **cardNumberCodriverSlotEnd** the generation 2 data structure makes use of the following data elements:

cardNumberAndGenDriverSlotBegin identifies the card including its generation which is inserted in the driver slot at the beginning of the event.

cardNumberAndGenCodriverSlotBegin identifies the card including its generation which is inserted in the co-driver slot at the beginning of the event.

cardNumberAndGenDriverSlotEnd identifies the card including its generation which is inserted in the driver slot at the end of the event.

cardNumberAndGenCodriverSlotEnd identifies the card including its generation which is inserted in the co-driver slot at the end of the event.

If the event is a time conflict the **eventBeginTime** and **eventEndTime** are to be interpreted as follows:

eventBeginTime is the ~~recording equipment~~ **control device** date and time.

eventEndTime is the GNSS date and time.

2.199. VuEventRecordArray

Generation 2:

Information, stored in a vehicle unit, related to events (~~Annex~~ **Appendix 1C** requirement 117 except over speeding event).

```
VuEventRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                 INTEGER(1..65535),
```

```

noOfRecords          INTEGER(0..65535),
records              SET SIZE(noOfRecords) OF VuEventRecord
}

```

recordType denotes the type of the record (VuEventRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuEventRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of events records.

2.200. VuFaultData

Generation 1:

Information, stored in a vehicle unit, related to faults (~~Annex~~ **Appendix 1B** requirement 096).

VuFaultData ::= SEQUENCE {

```

noOfVuFaults          INTEGER(0..255),
vuFaultRecords        SET SIZE(noOfVuFaults) OF VuFaultRecord
}

```

noOfVuFaults is the number of faults listed in the vuFaultRecords set.

vuFaultRecords is a set of faults records.

2.201. VuFaultRecord

Information, stored in a vehicle unit, related to a fault (~~Annex~~ **Appendix 1B** requirement 096 and ~~Annex~~ **Appendix 1C** requirement 118).

Generation 1:

VuFaultRecord ::= SEQUENCE {

```

faultType              EventFaultType,
faultRecordPurpose     EventFaultRecordPurpose,
faultBeginTime         TimeReal,
faultEndTime           TimeReal,
cardNumberDriverSlotBegin FullCardNumber,
cardNumberCodriverSlotBegin FullCardNumber,
cardNumberDriverSlotEnd FullCardNumber,
cardNumberCodriverSlotEnd FullCardNumber
}

```

faultType is the type of ~~recording equipment~~ **control device** fault.

faultRecordPurpose is the purpose for which this fault has been recorded.

faultBeginTime is the date and time of beginning of fault.

faultEndTime is the date and time of end of fault.

cardNumberDriverSlotBegin identifies the card inserted in the driver slot at the beginning of the fault.

cardNumberCodriverSlotBegin identifies the card inserted in the co-driver slot at the beginning of the fault.

cardNumberDriverSlotEnd identifies the card inserted in the driver slot at the end of the fault.

cardNumberCodriverSlotEnd identifies the card inserted in the co-driver slot at the end of the fault.

Generation 2:

```
VuFaultRecord ::= SEQUENCE {
    faultType                EventFaultType,
    faultRecordPurpose       EventFaultRecordPurpose,
    faultBeginTime           TimeReal,
    faultEndTime             TimeReal,
    cardNumberAndGenDriverSlotBegin FullCardNumberAndGeneration,
    cardNumberAndGenCodriverSlotBegin FullCardNumberAndGeneration,
    cardNumberAndGenDriverSlotEnd FullCardNumberAndGeneration,
    cardNumberAndGenCodriverSlotEnd FullCardNumberAndGeneration,
    manufacturerSpecificEventFaultData ManufacturerSpecificEventFaultData
}
```

In addition to generation 1 the following data element is used:

manufacturerSpecificEventFaultData contains additional, manufacturer specific information about the fault.

Instead of **cardNumberDriverSlotBegin**, **cardNumberCodriverSlotBegin**, **cardNumberDriverSlotEnd**, and **cardNumberCodriverSlotEnd** the generation 2 data structure makes use of the following data elements:

cardNumberAndGenDriverSlotBegin identifies the card including its generation which is inserted in the driver slot at the beginning of the fault.

cardNumberAndGenCodriverSlotBegin identifies the card including its generation which is inserted in the co-driver slot at the beginning of the fault.

cardNumberAndGenDriverSlotEnd identifies the card including its generation which is inserted in the driver slot at the end of the fault.

cardNumberAndGenCodriverSlotEnd identifies the card including its generation which is inserted in the co-driver slot at the end of the fault.

2.202. VuFaultRecordArray

Generation 2:

Information, stored in a vehicle unit, related to faults (~~Annex~~ **Appendix 1C** requirement 118).

```
VuFaultRecordArray ::= SEQUENCE {
```

recordType	RecordType,
recordSize	INTEGER(1..65535),
noOfRecords	INTEGER(0..65535),
records	SET SIZE(noOfRecords) OF VuFaultRecord
}	

recordType denotes the type of the record (VuFaultRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuFaultRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of faults records.

2.203. VuGNSSADRecord

Generation 2, **version 1:**

~~2.203. VuGNSSCDRecord~~

Information, stored in a vehicle unit, related to the GNSS position of the vehicle if the ~~continuous~~ **accumulated** driving time of the driver reaches a multiple of three hours (~~Annex~~ **Appendix 1C** requirement 108, 110).

~~VuGNSSCDRecord~~ **VuGNSSADRecord ::= SEQUENCE {**

timeStamp	TimeReal,
cardNumberAndGenDriverSlot	FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlot	FullCardNumberAndGeneration,
gnssPlaceRecord	GNSSPlaceRecord,
vehicleOdometerValue	OdometerShort
}	

timeStamp is the date and time when the ~~continuous~~ **accumulated** driving time of the card holder reaches a multiple of three hours.

cardNumberAndGenDriverSlot identifies the card including its generation which is inserted in the driver slot.

cardNumberAndGenCodriverSlot identifies the card including its generation which is inserted in the co-driver slot.

gnssPlaceRecord contains information related to the position of the vehicle.

vehicleOdometerValue is the odometer value when the accumulated driving time reaches a multiple of three hours.

Generation 2, **version 2:**

Information, stored in a vehicle unit, related to the GNSS position of the vehicle if the accumulated driving time reaches a multiple of three hours (**Appendix 1C** requirement 108, 110).

VuGNSSADRecord ::= SEQUENCE {

timestamp	TimeReal,
------------------	------------------

```

cardNumberAndGenDriverSlot FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlot FullCardNumberAndGeneration,
gnssPlaceAuthRecord GNSSPlaceAuthRecord,
vehicleOdometerValue OdometerShort
}

```

In Generation 2 version 2, instead of `gnssPlaceRecord`, the `gnssPlaceAuthRecord` is used, which contains the GNSS authentication status in addition.

2.203a. VuBorderCrossingRecord

Generation 2, version 2:

Information, stored in a vehicle unit, related to border crossings of the vehicle when the latter has crossed the border of a country (Appendix 1C requirement 133a and 133b).

```

VuBorderCrossingRecord ::= SEQUENCE {
    cardNumberAndGenDriverSlot FullCardNumberAndGeneration,
    cardNumberAndGenCodriverSlot FullCardNumberAndGeneration,
    countryLeft NationNumeric,
    countryEntered NationNumeric,
    gnssPlaceAuthRecord GNSSPlaceAuthRecord,
    vehicleOdometerValue OdometerShort
}

```

`cardNumberAndGenDriverSlot` identifies the card including its generation which is inserted in the driver slot.

`cardNumberAndGenCodriverSlot` identifies the card including its generation which is inserted in the co-driver slot.

`countryLeft` is the country which was left by the vehicle, based on the last available position before the border crossing has been detected. ‘Rest of the World’ (NationNumeric code ‘FF’H) shall be used when the vehicle unit is not able to determine the country where the vehicle is located (e.g. the current country is not part of the stored digital maps).

`countryEntered` is the country into which the vehicle has entered. ‘Rest of the World’ (NationNumeric code ‘FF’H) shall be used when the vehicle unit is not able to determine the country where the vehicle is located (e.g. the current country is not part of the stored digital maps).

`gnssPlaceAuthRecord` contains information related to the position of the vehicle when the border crossing was detected, and its authentication status.

`vehicleOdometerValue` is the odometer value when the vehicle unit has detected that the vehicle has crossed the border of a country.

2.203b. VuBorderCrossingRecordArray

Generation 2, version 2:

Information, stored in a vehicle unit, related to border crossings of the vehicle (Appendix 1C requirement 133c).

```
VuBorderCrossingRecordArray ::= SEQUENCE {
  recordType      RecordType,
  recordSize      INTEGER(1..65535),
  noOfRecords     INTEGER(0..65535),
  records         SET SIZE(noOfRecords) OF VuBorderCrossingRecord
}
```

recordType denotes the type of the record (VuBorderCrossingRecord). Value assignment: see RecordType.

recordSize is the size of the VuBorderCrossingRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of border crossing records.

2.204. VuGNSSADRecordArray

~~2.204. VuGNSSCDRecordArray~~

Generation 2:

Information, stored in a vehicle unit, related to the GNSS position of the vehicle if the ~~continuous~~ **accumulated** driving time of the driver reaches a multiple of three hours (~~Annex Appendix 1C requirement 108 and 110~~).

```
VuGNSSADRecordArray ::= SEQUENCE {
  recordType      RecordType,
  recordSize      INTEGER(1..65535),
  noOfRecords     INTEGER(0..65535),
  records         SET SIZE (noOfRecords) OF
VuGNSSCDRecord      VuGNSSADRecord
}
```

recordType denotes the type of the record ~~VuGNSSCDRecord~~ (**VuGNSSADRecord**).

Value Assignment: See RecordType

recordSize is the size of the ~~VuGNSSCDRecord~~ **VuGNSSADRecord** in bytes.

noOfRecords is the number of records in the set records.

records is a set of GNSS ~~continuous~~ **accumulated** driving records.

2.204a. VuGnssMaximalTimeDifference

Generation 2, version 2:

The maximal difference between true time and the VU Real Time Clock time, based on the maximal time drift specified in Appendix 1C requirement 041, transmitted by the vehicle unit to an external GNSS Facility, see Sub-appendix 12 requirement GNS_3g.

VuGnssMaximalTimeDifference ::= INTEGER(0..65535)

2.205. VuIdentification

Information, stored in a vehicle unit, related to the identification of the vehicle unit (~~Annex~~ **Appendix 1B** requirement 075 and ~~Annex~~ **Appendix 1C** requirement 93 and 121).

Generation 1:

```
VuIdentification ::= SEQUENCE {
vuManufacturerName          VuManufacturerName,
vuManufacturerAddress       VuManufacturerAddress,
vuPartNumber                VuPartNumber,
vuSerialNumber              VuSerialNumber,
vuSoftwareIdentification    VuSoftwareIdentification,
vuManufacturingDate         VuManufacturingDate,
vuApprovalNumber            VuApprovalNumber
}
```

vuManufacturerName is the name of the manufacturer of the vehicle unit.

vuManufacturerAddress is the address of the manufacturer of the vehicle unit.

vuPartNumber is the part number of the vehicle unit.

vuSerialNumber is the serial number of the vehicle unit.

vuSoftwareIdentification identifies the software implemented in the vehicle unit.

vuManufacturingDate is the manufacturing date of the vehicle unit.

vuApprovalNumber is the type approval number of the vehicle unit.

Generation 2:

```
VuIdentification ::= SEQUENCE {
vuManufacturerName          VuManufacturerName,
vuManufacturerAddress       VuManufacturerAddress,
vuPartNumber                VuPartNumber,
vuSerialNumber              VuSerialNumber,
vuSoftwareIdentification    VuSoftwareIdentification,
vuManufacturingDate         VuManufacturingDate,
vuApprovalNumber            VuApprovalNumber,
vuGeneration                Generation,
vuAbility                   VuAbility
vuDigitalMapVersion       VuDigitalMapVersion
}
```

In addition to generation 1 the following data elements are used:

vuGeneration identifies the generation of the vehicle unit.

vuAbility provides information whether the VU supports generation 1 tachograph cards or not.

vuDigitalMapVersion is the version of the digital map stored in the vehicle unit (only present in version 2).

2.206. VuIdentificationRecordArray

Generation 2:

The VuIdentification plus metadata used in the download protocol.

```
VuIdentificationRecordArray ::= SEQUENCE {  
    recordType          RecordType,  
    recordSize          INTEGER(1..65535),  
    noOfRecords        INTEGER(0..65535),  
    records             SET SIZE(noOfRecords) OF VuIdentification  
}
```

recordType denotes the type of the record (VuIdentification). **Value Assignment:** See RecordType

recordSize is the size of the VuIdentification in bytes.

noOfRecords is the number of records in the set records.

records is a set of VuIdentification records.

2.207. VuITSConsentRecord

Generation 2:

Information stored in a vehicle unit, related to the consent of a driver to use Intelligent Transport Systems.

```
VuITSConsentRecord ::= SEQUENCE {  
    cardNumberAndGen    FullCardNumberAndGeneration,  
    consent              BOOLEAN  
}
```

cardNumberAndGen identifies the card including its generation. This must be a driver card or a workshop card.

consent is a flag which indicates whether the driver has given his consent on the usage of Intelligent Transport Systems with this vehicle / vehicle unit.

Value assignment:

TRUE indicates the driver's consent to use Intelligent Transport Systems

FALSE indicates the driver's denial to use Intelligent Transport Systems

2.208. VuITSConsentRecordArray

Generation 2:

Information, stored in a vehicle unit, related to drivers' consent on the usage of Intelligent Transport Systems (~~Annex~~ **Appendix 1C** requirement 200).

```
VuITSConsentRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET SIZE(noOfRecords) OF VuITSConsentRecord
}
```

recordType denotes the type of the record (VuITSConsentRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuITSConsentRecord in bytes.

noOfRecords is the number of records in the set records.

records is the set of ITS consent records.

2.208a VuLoadUnloadRecord

Generation 2, version 2:

Information, stored in the vehicle unit, related to a load/unload operation entered (**Appendix 1C** requirements 133e, 133f and 133g).

```
VuLoadUnloadRecord ::= SEQUENCE {
timeStamp                TimeReal,
operationType            OperationType
cardNumberAndGenDriverSlot FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlot FullCardNumberAndGeneration,
gnssPlaceAuthRecord     GNSSPlaceAuthRecord,
vehicleOdometerValue     OdometerShort
}
```

timeStamp is the date and time when the load/unload operation was entered.

operationType is the type of the operation entered (load, unload, or simultaneous load/unload).

cardNumberAndGenDriverSlot identifies the card including its generation which is inserted in the driver slot.

cardNumberAndGenCodriverSlot identifies the card including its generation which is inserted in the co-driver slot.

gnssPlaceAuthRecord contains information related to the position of the vehicle, and its authentication status.

vehicleOdometerValue is the odometer value related to the load/unload operation.

2.208b VuLoadUnloadRecordArray

Generation 2, version 2:

Information, stored in a vehicle unit, related to a load/unload operation vehicle entered (Appendix 1C requirement 133h).

```
VuLoadUnloadRecordArray ::= SEQUENCE {
    recordType          RecordType,
    recordSize          INTEGER(1..65535),
    noOfRecords         INTEGER(0..65535),
    records             SET          SIZE(noOfRecords)      OF
                        VuLoadUnloadRecord
}
```

recordType denotes the type of the record (VuLoadUnloadRecord). **Value Assignment:** See Record Type.

recordSize is the size of the VuLoadUnloadRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of load/unload operation records.

2.209. VuManufacturerAddress

Address of the manufacturer of the vehicle unit.

VuManufacturerAddress ::= Address

Value assignment: Unspecified.

2.210. VuManufacturerName

Name of the manufacturer of the vehicle unit.

VuManufacturerName ::= Name

Value assignment: Unspecified.

2.211. VuManufacturingDate

Date of manufacture of the vehicle unit.

VuManufacturingDate ::= TimeReal

Value assignment: Unspecified.

2.212. VuOverSpeedingControlData

Information, stored in a vehicle unit, related to over speeding events since the last over speeding control (~~Annex~~ **Appendix** 1B requirement 095 and ~~Annex~~ **Appendix** 1C requirement 117).

```
VuOverSpeedingControlData ::= SEQUENCE {
```

```

lastOverspeedControlTime    TimeReal,
firstOverspeedSince         TimeReal,
numberOfOverspeedSince      OverspeedNumber
}

```

lastOverspeedControlTime is the date and time of the last over speeding control.

firstOverspeedSince is the date and time of the first over speeding following this over speeding control.

numberOfOverspeedSince is the number of over speeding events since the last over speeding control.

2.213. VuOverSpeedingControlDataRecordArray

Generation 2:

The VuOverSpeedingControlData plus metadata used in the download protocol.

```

VuOverSpeedingControlDataRecordArray ::= SEQUENCE {
recordType                    RecordType,
recordSize                    INTEGER(1..65535),
noOfRecords                   INTEGER(0..65535),
records                       SET SIZE(noOfRecords) OF
                              VuOverSpeedingControlData
}

```

recordType denotes the type of the record (VuOverSpeedingControlData). **Value Assignment:** See RecordType

recordSize is the size of the VuOverSpeedingControlData in bytes.

noOfRecords is the number of records in the set records.

records is a set of over speeding control data records.

2.214. VuOverSpeedingEventData

Generation 1:

Information, stored in a vehicle unit, related to over speeding events (~~Annex~~ **Appendix 1B** requirement 094).

```

VuOverSpeedingEventData ::= SEQUENCE {
noOfVuOverSpeedingEvents     INTEGER(0..255),
vuOverSpeedingEventRecords   SET SIZE(noOfVuOverSpeedingEvents) OF
                              VuOverSpeedingEventRecord
}

```

noOfVuOverSpeedingEvents is the number of events listed in the vuOverSpeedingEventRecords set.

vuOverSpeedingEventRecords is a set of over speeding events records.

2.215. VuOverSpeedingEventRecord

Generation 1:

Information, stored in a vehicle unit, related to over speeding events (~~Annex~~ **Appendix 1B** requirement 094 and ~~Annex~~ **Appendix 1C** requirement 117).

```
VuOverSpeedingEventRecord ::= SEQUENCE {
eventType                EventFaultType,
eventRecordPurpose       EventFaultRecordPurpose,
eventBeginTime           TimeReal,
eventEndTime             TimeReal,
maxSpeedValue            SpeedMax,
averageSpeedValue        SpeedAverage,
cardNumberDriverSlotBegin FullCardNumber,
similarEventsNumber      SimilarEventsNumber
}
```

eventType is the type of the event.

eventRecordPurpose is the purpose for which this event has been recorded.

eventBeginTime is the date and time of beginning of event.

eventEndTime is the date and time of end of event.

maxSpeedValue is the maximum speed measured during the event.

averageSpeedValue is the arithmetic average speed measured during the event.

cardNumberDriverSlotBegin identifies the card inserted in the driver slot at the beginning of the event.

similarEventsNumber is the number of similar events that day.

Generation 2:

Information, stored in a vehicle unit, related to over speeding events (~~Annex~~ **Appendix 1B** requirement 094 and ~~Annex~~ **Appendix 1C** requirement 117).

```
VuOverSpeedingEventRecord ::= SEQUENCE {
eventType                EventFaultType,
eventRecordPurpose       EventFaultRecordPurpose,
eventBeginTime           TimeReal,
eventEndTime             TimeReal,
maxSpeedValue            SpeedMax,
averageSpeedValue        SpeedAverage,
cardNumberAndGenDriverSlotBegin FullCardNumberAndGeneration,
similarEventsNumber      SimilarEventsNumber
}
```

Instead of `cardNumberDriverSlotBegin`, the generation 2 data structure makes use of the following data element:

cardNumberAndGenDriverSlotBegin identifies the card including its generation which is inserted in the driver slot at the beginning of the event.

2.216. VuOverSpeedingEventRecordArray

Generation 2:

Information, stored in a vehicle unit, related to over speeding events (~~Annex~~ **Appendix 1C** requirement 117).

```
VuOverSpeedingEventRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET          SIZE(noOfRecords)          OF
                          VuOverSpeedingEventRecord
}
```

recordType denotes the type of the record (`VuOverSpeedingEventRecord`). **Value Assignment:** See `RecordType`

recordSize is the size of the `VuOverSpeedingEventRecord` in bytes.

noOfRecords is the number of records in the set `records`.

records is a set of over speeding events records.

2.217. VuPartNumber

Part number of the vehicle unit.

```
VuPartNumber ::= IA5String(SIZE(16))
```

Value assignment: VU manufacturer specific.

2.218. VuPlaceDailyWorkPeriodData

Generation 1:

Information, stored in a vehicle unit, related to places where drivers begin or end a daily work period (~~Annex~~ **Appendix 1B** requirement 087 and ~~Annex~~ **Appendix 1C** requirement 108 and 110).

```
VuPlaceDailyWorkPeriodData ::= SEQUENCE {
noOfPlaceRecords          INTEGER(0..255),
vuPlaceDailyWorkPeriodRecords SET SIZE(noOfPlaceRecords) OF
                          VuPlaceDailyWorkPeriodRecord
}
```

noOfPlaceRecords is the number of records listed in the `vuPlaceDailyWorkPeriodRecords` set.

vuPlaceDailyWorkPeriodRecords is a set of place related records.

2.219. VuPlaceDailyWorkPeriodRecord

Generation 1:

Information, stored in a vehicle unit, related to a place where a driver begins or ends a daily work period (~~Annex~~ **Appendix 1B** requirement 087 and ~~Annex~~ **Appendix 1C** requirement 108 and 110).

```
VuPlaceDailyWorkPeriodRecord ::= SEQUENCE {
    fullCardNumber          FullCardNumber,
    placeRecord             PlaceRecord
}
```

fullCardNumber is the driver's card type, card issuing ~~Member State~~ **Contracting Party** and card number.

placeRecord contains the information related to the place entered.

Generation 2, **version 1**:

Information, stored in a vehicle unit, related to a place where a driver begins or ends a daily work period (~~Annex~~ **Appendix 1B** requirement 087 and ~~Annex~~ **Appendix 1C** requirement 108 and 110).

```
VuPlaceDailyWorkPeriodRecord ::= SEQUENCE {
    fullCardNumberAndGeneration FullCardNumberAndGeneration,
    placeRecord                 PlaceRecord
}
```

Instead of fullCardNumber, the generation 2 data structure makes use of the following data element:

fullCardNumberAndGeneration is the type of card, its issuing ~~Member State~~ **Contracting Party**, its card number and generation as stored in the card.

Generation 2, **version 2**:

Information, stored in a vehicle unit, related to a place where a driver begins or ends a daily work period (Appendix 1B requirement 087 and Appendix 1C requirement 108 and 110).

```
VuPlaceDailyWorkPeriodRecord ::= SEQUENCE {
    fullCardNumberAndGeneration FullCardNumberAndGeneration,
    placeAuthRecord             PlaceAuthRecord
}
```

Instead of placeRecord, the generation 2 version 2 data structure makes use of the following data element:

placeAuthRecord contains the information related to the place entered, the recorded position, GNSS authentication status and position determination time.

2.220. VuPlaceDailyWorkPeriodRecordArray

Generation 2:

Information, stored in a vehicle unit, related to places where drivers begin or end a daily work period (~~Annex~~ **Appendix** 1C requirement 108 and 110).

```
VuPlaceDailyWorkPeriodRecordArray ::= SEQUENCE {
recordType          RecordType,
recordSize          INTEGER(1..65535),
noOfRecords         INTEGER(0..65535),
records             SET SIZE(noOfRecords) OF
                   VuPlaceDailyWorkPeriodRecord
}
```

recordType denotes the type of the record (VuPlaceDailyWorkPeriodRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuPlaceDailyWorkPeriodRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of place related records.

2.221. VuPrivateKey

Generation 1:

The private key of a vehicle unit.

```
VuPrivateKey ::= RSAKeyPrivateExponent
```

2.222. VuPublicKey

Generation 1:

The public key of a vehicle unit.

```
VuPublicKey ::= PublicKey
```

2.222a. VuRtcTime

Generation 2, version 2:

The time of the VU RTC clock, transmitted by the VU to an External GNSS Facility, see Sub-appendix 12 requirement GNS_3f.

```
VuRtcTime ::= TimeReal
```

2.223. VuSerialNumber

Serial number of the vehicle unit (~~Annex~~ **Appendix** 1B requirement 075 and ~~Annex~~ **Appendix** 1C requirement 93).

```
VuSerialNumber ::= ExtendedSerialNumber
```

2.224. VuSoftInstallationDate

Date of installation of the vehicle unit software version.

VuSoftInstallationDate ::= TimeReal

Value assignment: Unspecified.

2.225. VuSoftwareIdentification

Information, stored in a vehicle unit, related to the software installed.

VuSoftwareIdentification ::= SEQUENCE {

vuSoftwareVersion VuSoftwareVersion,

vuSoftInstallationDate VuSoftInstallationDate

}

vuSoftwareVersion is the software version number of the Vehicle Unit.

vuSoftInstallationDate is the software version installation date.

2.226. VuSoftwareVersion

Software version number of the vehicle unit.

VuSoftwareVersion ::= IA5String(SIZE(4))

Value assignment: Unspecified.

2.227. VuSpecificConditionData

Generation 1:

Information, stored in a vehicle unit, related to specific conditions.

VuSpecificConditionData ::= SEQUENCE {

noOfSpecificConditionRecords INTEGER(0..2¹⁶-1)

specificConditionRecords SET SIZE (noOfSpecificConditionRecords) OF SpecificConditionRecord

}

noOfSpecificConditionRecords is the number of records listed in the specificConditionRecords set.

specificConditionRecords is a set of specific conditions related records.

2.228. VuSpecificConditionRecordArray

Generation 2:

Information, stored in a vehicle unit, related to specific conditions (~~Annex~~ **Appendix 1C** requirement 130).

VuSpecificConditionRecordArray ::= SEQUENCE {

recordType	RecordType,
recordSize	INTEGER(1..65535),
noOfRecords	INTEGER(0..65535),
records	SET SIZE(noOfRecords) OF SpecificConditionRecord
	}

recordType denotes the type of the record (SpecificConditionRecord). **Value Assignment:** See RecordType

recordSize is the size of the SpecificConditionRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of specific conditions related records.

2.229. VuTimeAdjustmentData

Generation 1:

Information, stored in a vehicle unit, related to time adjustments performed outside the frame of a regular calibration (~~Annex~~ **Appendix 1B** requirement 101).

VuTimeAdjustmentData ::= SEQUENCE {

noOfVuTimeAdjRecords	INTEGER(0..6),
----------------------	----------------

vuTimeAdjustmentRecords	SET SIZE(noOfVuTimeAdjRecords) OF VuTimeAdjustmentRecord
-------------------------	---

}

noOfVuTimeAdjRecords is the number of records in vuTimeAdjustmentRecords.

vuTimeAdjustmentRecords is a set of time adjustment records.

2.230. Reserved for future use

~~2.230. VuTimeAdjustmentGNSSRecord~~

2.231. Reserved for future use

2.232. VuTimeAdjustmentRecord

~~Information, stored in a vehicle unit, related to a time adjustment based on time data from GNSS (Annex ~~IC~~ requirement 124 and 125)~~

~~VuTimeAdjustmentGNSSRecord SEQUENCE~~

oldTimeValue,	TimeReal
--------------------------	---------------------

newTimeValue	TimeReal
-------------------------	---------------------

~~OldTimeVaule, newTimeVaule are the old and new values of date and time~~

~~2.231 VuTimeAdjustmentGNSSRecordArray~~

Information, stored in a vehicle unit, related to a time adjustment based on time data from GNSS (Annex IC requirement 124 and 125).

~~VuTimeAdjustmentGNSSRecordArray SEQUENCE~~

~~recordType RecordType~~

~~recordSize INTEGER(1.65535)~~

~~noOfRecords INTEGER(0.65535)~~

~~records SETSIZE(noOfRecords)OFVuTimeAdjustmentGNSSRecord~~

~~recordType denotes the type of the record (VuTimeAdjustmentGNSSRecord). Value Assignment: See RecordType~~

~~recordSize is the size of the VuTimeAdjustmentGNSSRecord in bytes~~

~~noOfRecords is the number of records in the set records;~~

~~records is a set of GNSS time adjustment records.~~

Information, stored in a vehicle unit, related a time adjustment performed outside the frame of a regular calibration (Annex Appendix 1B requirement 101 and Annex Appendix 1C requirement 124 and 125).

Generation 1:

VuTimeAdjustmentRecord ::= SEQUENCE {

oldTimeValue TimeReal,

newTimeValue TimeReal,

workshopName Name,

workshopAddress Address,

workshopCardNumber FullCardNumber

}

oldTimeValue, **newTimeValue** are the old and new values of date and time.

workshopName, **workshopAddress** are the workshop name and address.

workshopCardNumber identifies the workshop card used to perform the time adjustment.

Generation 2:

VuTimeAdjustmentRecord ::= SEQUENCE {

oldTimeValue TimeReal,

newTimeValue TimeReal,

workshopName Name,

workshopAddress Address,

workshopCardNumberAndGeneration FullCardNumberAndGeneration

}

Instead of workshopCardNumber the generation 2 data structure makes use of the following data element.

workshopCardNumberAndGeneration identifies the workshop card including its generation used to perform the time adjustment.

2.233. VuTimeAdjustmentRecordArray

Generation 2:

Information, stored in a vehicle unit, related to time adjustments performed outside the frame of a regular calibration (~~Annex~~ **Appendix 1C** requirement 124 and 125).

```
VuTimeAdjustmentRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET SIZE(noOfRecords) OF
                          VuTimeAdjustmentRecord
}
```

recordType denotes the type of the record (VuTimeAdjustmentRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuTimeAdjustmentRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of time adjustment records.

2.234. WorkshopCardApplicationIdentification

Information, stored in a workshop card related to the identification of the application of the card (~~Annex~~ **Appendix 1C** requirement 307 and 330).

Generation 1:

```
WorkshopCardApplicationIdentification ::= SEQUENCE {
typeOfTachographCardId   EquipmentType,
cardStructureVersion      CardStructureVersion,
noOfEventsPerType         NoOfEventsPerType,
noOfFaultsPerType         NoOfFaultsPerType,
activityStructureLength    CardActivityLengthRange,
noOfCardVehicleRecords    NoOfCardVehicleRecords,
noOfCardPlaceRecords      NoOfCardPlaceRecords,
noOfCalibrationRecords    NoOfCalibrationRecords
}
```

typeOfTachographCardId is specifying the implemented type of card.

cardStructureVersion is specifying the the version of the structure that is implemented in the card.

noOfEventsPerType is the number of events per type of event the card can record.

noOfFaultsPerType is the number of faults per type of fault the card can record.

activityStructureLength indicates the number of bytes available for storing activity records.

noOfCardVehicleRecords is the number of vehicle records the card can contain.

noOfCardPlaceRecords is the number of places the card can record.

noOfCalibrationRecords is the number of calibration records the card can store.

Generation 2:

```
WorkshopCardApplicationIdentification ::= SEQUENCE {
  typeOfTachographCardId      EquipmentType,
  cardStructureVersion         CardStructureVersion,
  noOfEventsPerType           NoOfEventsPerType,
  noOfFaultsPerType           NoOfFaultsPerType,
  activityStructureLength      CardActivityLengthRange,
  noOfCardVehicleRecords      NoOfCardVehicleRecords,
  noOfCardPlaceRecords        NoOfCardPlaceRecords,
  noOfCalibrationRecords      NoOfCalibrationRecords,
  noOfGNSSCDRecords       NoOfGNSSCDRecords,
  noOfGNSSADRecords        NoOfGNSSADRecords,
  noOfSpecificConditionRecords NoOfSpecificConditionRecords,
  noOfCardVehicleUnitRecords NoOfCardVehicleUnitRecords
}
```

In addition to generation 1 the following data elements are used:

~~noOfGNSSCDRecords~~ **noOfGNSSADRecords** is the number of GNSS ~~continuous~~ **accumulated** driving records the card can store.

noOfSpecificConditionRecords is the number of specific condition records the card can store.

noOfCardVehicleUnitRecords is the number of vehicle units used records the card can store.

2.234a. WorkshopCardApplicationIdentificationV2

Generation 2, version 2:

Information, stored in a workshop card related to the identification of the application of the card (Appendix 1C requirement 330a).

WorkshopCardApplicationIdentificationV2 ::= SEQUENCE {

lengthOfFollowingData	LengthOfFollowingData,
noOfBorderCrossingRecords	NoOfBorderCrossingRecords,
noOfLoadUnloadRecords	NoOfLoadUnloadRecords,
noOfLoadTypeEntryRecords	NoOfLoadTypeEntryRecords,
vuConfigurationLengthRange	VuConfigurationLengthRange

}

lengthOfFollowingData is the number of bytes following in the record.

noOfBorderCrossingRecords is the number of border crossing records the workshop card can store.

noOfLoadUnloadRecords is the number of load/unload records the workshop card can store.

noOfLoadTypeEntryRecords is the number of load type entry records the workshop card can store.

vuConfigurationLengthRange is the number of bytes in a tachograph card, available to store VU configurations.

2.234b. WorkshopCardCalibrationAddData

Generation 2, version 2:

Information, stored in a workshop card, related to the additional data (i.e. by-default load type) entered during a calibration (Appendix 1C requirement 356l).

WorkshopCardCalibrationAddData ::= SEQUENCE {

calibrationPointerNewestRecord	INTEGER(0..NoOfCalibrationRecords -1),
workshopCardCalibrationAddDataRecords	SET SIZE(NoOfCalibrationRecords) OF WorkshopCardCalibrationAddDataRecord

}

calibrationPointerNewestRecord is the index of the last updated calibration additional data record.

Value assignment is the number corresponding to the numerator of the calibration additional data record, beginning with '0' for the first occurrence of the calibration additional data record in the structure.

workshopCardCalibrationAddDataRecords is the set of records containing the old date and time value, the vehicle identification value and the by-default load type of the vehicle.

2.234c. WorkshopCardCalibrationAddDataRecord

Generation 2, version 2:

Information, stored in a workshop card, related to the by-default load type entered during a calibration (Appendix 1C requirement 356k).

WorkshopCardCalibrationAddDataRecord ::= SEQUENCE {

oldTimeValue **TimeReal,**
vehicleIdentificationNumber **VehicleIdentificationNumber,**
byDefaultLoadType **LoadType,**
calibrationCountry **NationNumeric,**
calibrationCountryTimestamp **TimeReal**
}

oldTimeValue is the old value of date and time contained in the corresponding **WorkshopCardCalibrationRecord**,

vehicleIdentificationNumber is the vehicle identification number of the vehicle, also contained in the corresponding **WorkshopCardCalibrationRecord**,

byDefaultLoadType is the by-default load type of the vehicle (only present in version 2).

calibrationCountry is the country in which the calibration has been performed,

calibrationCountryTimestamp is the date time when the position used to determine this country was provided by the GNSS receiver.

2.235. WorkshopCardCalibrationData

Information, stored in a workshop card, related to workshop activity performed with the card (~~Annex~~ **Appendix 1C** requirements 314, 316, 337, and 339).

WorkshopCardCalibrationData ::= SEQUENCE {

calibrationTotalNumber **INTEGER(0 .. 2¹⁶-1),**
calibrationPointerNewestRecord **INTEGER(0 .. NoOfCalibrationRecords-1),**
calibrationRecords **SET SIZE(NoOfCalibrationRecords) OF**
WorkshopCardCalibrationRecord
}

calibrationTotalNumber is the total number of calibrations performed with the card.

calibrationPointerNewestRecord is the index of the last updated calibration record.

Value assignment: Number corresponding to the numerator of the calibration record, beginning with '0' for the first occurrence of the calibration records in the structure.

calibrationRecords is the set of records containing calibration and/or time adjustment information.

2.236. WorkshopCardCalibrationRecord

Information, stored in a workshop card, related to a calibration performed with the card (~~Annex~~ **Appendix 1C** requirement 314 and 337).

Generation 1:

WorkshopCardCalibrationRecord ::= SEQUENCE {

calibrationPurpose **CalibrationPurpose,**

vehicleIdentificationNumber	VehicleIdentificationNumber,
vehicleRegistration	VehicleRegistrationIdentification,
wVehicleCharacteristicConstant	W-VehicleCharacteristicConstant,
kConstantOfRecordingEquipment	K-ConstantOfRecordingEquipment,
lTyreCircumference	L-TyreCircumference,
tyreSize	TyreSize,
authorisedSpeed	SpeedAuthorised,
oldOdometerValue	OdometerShort,
newOdometerValue	OdometerShort,
oldTimeValue	TimeReal,
newTimeValue	TimeReal,
nextCalibrationDate	TimeReal,
vuPartNumber	VuPartNumber,
vuSerialNumber	VuSerialNumber,
sensorSerialNumber	SensorSerialNumber
}	

calibrationPurpose is the purpose of the calibration.

vehicleIdentificationNumber is the VIN.

vehicleRegistration contains the VRN and registering ~~Member State~~ **Contracting Party**.

wVehicleCharacteristicConstant is the characteristic coefficient of the vehicle.

kConstantOfRecordingEquipment is the constant of the ~~recording equipment~~ **control device**.

lTyreCircumference is the effective circumference of the wheel tyres.

tyreSize is the designation of the dimensions of the tyres mounted on the vehicle.

authorisedSpeed is the maximum authorised speed of the vehicle.

oldOdometerValue, **newOdometerValue** are the old and new values of the odometer.

oldTimeValue, **newTimeValue** are the old and new values of date and time.

nextCalibrationDate is the date of the next calibration of the type specified in CalibrationPurpose to be carried out by the authorised inspection authority.

vuPartNumber, **vuSerialNumber** and **sensorSerialNumber** are the data elements for ~~recording equipment~~ **control device** identification.

Generation 2:

WorkshopCardCalibrationRecord ::= SEQUENCE {

calibrationPurpose	CalibrationPurpose,
vehicleIdentificationNumber	VehicleIdentificationNumber,
vehicleRegistration	VehicleRegistrationIdentification,
wVehicleCharacteristicConstant	W-VehicleCharacteristicConstant,

kConstantOfRecordingEquipment	K-ConstantOfRecordingEquipment,
lTyreCircumference	L-TyreCircumference,
tyreSize	TyreSize,
authorisedSpeed	SpeedAuthorised,
oldOdometerValue	OdometerShort,
newOdometerValue	OdometerShort,
oldTimeValue	TimeReal,
newTimeValue	TimeReal,
nextCalibrationDate	TimeReal,
vuPartNumber	VuPartNumber,
vuSerialNumber	VuSerialNumber,
sensorSerialNumber	SensorSerialNumber,
sensorGNSSSerialNumber	SensorGNSSSerialNumber,
rcmSerialNumber	RemoteCommunicationModuleSerialNumber,
sealDataCard	SealDataCard
}	

In addition to generation 1 the following data elements are used:

sensorGNSSSerialNumber which identifies an external GNSS facility.

rcmSerialNumber which identifies a Remote Communication Module.

sealDataCard gives information about the seals that are attached to different components of the vehicle.

2.237. WorkshopCardHolderIdentification

Information, stored in a workshop card, related to the identification of the cardholder (~~Annex~~ **Appendix 1C** requirement 311 and 334).

```
WorkshopCardHolderIdentification ::= SEQUENCE {
workshopName          Name,
workshopAddress       Address,
cardHolderName       HolderName,
cardHolderPreferredLanguage Language
}
```

workshopName is name of the workshop of the card holder.

workshopAddress is the address of the workshop of the card holder.

cardHolderName is the name and first name(s) of the holder (e.g. the name of the mechanic).

cardHolderPreferredLanguage is the preferred language of the card holder.

2.238. WorkshopCardPIN

Personal identification number of the Workshop Card (~~Annex~~ **Appendix 1C** requirement 309 and 332).

WorkshopCardPIN ::= IA5String(SIZE(8))

Value assignment: The PIN known to the cardholder, right padded with 'FF' bytes up to 8 bytes.

2.239. W-VehicleCharacteristicConstant

Characteristic coefficient of the vehicle (definition k).

W-VehicleCharacteristicConstant ::= INTEGER(0..2¹⁶-1)

Value assignment: Impulses per kilometer in the operating range 0 to 64 255 pulses/km.

2.240. VuPowerSupplyInterruptionRecord

Generation 2:

Information, stored in a vehicle unit, related to Power Supply Interruption events (~~Annex~~ **Appendix 1C** requirement 117).

```
VuPowerSupplyInterruptionRecord ::= SEQUENCE {
eventType                               EventFaultType,
eventRecordPurpose                       EventFaultRecordPurpose,
eventBeginTime                           TimeReal,
eventEndTime                             TimeReal,
cardNumberAndGenDriverSlotBegin         FullCardNumberAndGeneration,
cardNumberAndGenDriverSlotEnd           FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlotBegin       FullCardNumberAndGeneration,
cardNumberAndGenCodriverSlotEnd         FullCardNumberAndGeneration,
similarEventsNumber                       SimilarEventsNumber
}
```

eventType is the type of the event.

eventRecordPurpose is the purpose for which this event has been recorded.

eventBeginTime is the date and time of beginning of event.

eventEndTime is the date and time of end of event.

cardNumberAndGenDriverSlotBegin identifies the card including its generation inserted in the driver slot at the beginning of the event.

cardNumberAndGenDriverSlotEnd identifies the card including its generation inserted in the driver slot at the end of the event.

cardNumberAndGenCodriverSlotBegin identifies the card including its generation inserted in the co-driver slot at the beginning of the event.

cardNumberAndGenCodriverSlotEnd identifies the card including its generation inserted in the co-driver slot at the end of the event.

similarEventsNumber is the number of similar events that day.

2.241. VuPowerSupplyInterruptionRecordArray

Generation 2:

Information, stored in a vehicle unit, related to Power Supply Interruption events (~~Annex~~ **Appendix** 1C requirement 117).

```
VuPowerSupplyInterruptionRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535)
records                   SET SIZE (noOfRecords) OF
                          VuPowerSupplyInterruptionRecord
}
```

recordType denotes the type of the record (VuPowerSupplyInterruptionRecord). **Value Assignment:** See RecordType

recordSize is the size of the VuPowerSupplyInterruptionRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of power supply interruption events records.

2.242. VuSensorExternalGNSSCoupledRecordArray

Generation 2:

A set of SensorExternalGNSSCoupledRecord plus metadata used in the download protocol.

```
VuSensorExternalGNSSCoupledRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords              INTEGER(0..65535),
records                   SET      SIZE(noOfRecords)      OF
                          SensorExternalGNSSCoupledRecord
}
```

recordType denotes the type of the record (SensorExternalGNSSCoupledRecord). **Value Assignment:** See RecordType

recordSize is the size of the SensorExternalGNSSCoupledRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of Sensor External GNSS Coupled records.

2.243. VuSensorPairedRecordArray

Generation 2:

A set of SensorPairedRecord plus metadata used in the download protocol.

```
VuSensorPairedRecordArray ::= SEQUENCE {
recordType                RecordType,
recordSize                INTEGER(1..65535),
noOfRecords               INTEGER(0..65535),
records                   SET      SIZE(noOfRecords)    OF
SensorPairedRecord
}
```

recordType denotes the type of the record (SensorPairedRecord). **Value Assignment:** See RecordType

recordSize is the size of the SensorPairedRecord in bytes.

noOfRecords is the number of records in the set records.

records is a set of sensor paired records.

3. Value and size range definitions

Definition of variable values used for definitions in paragraph 2.

TimeRealRange ::= $2^{32}-1$

4. Character sets

IA5Strings use the ASCII characters as defined by ISO/IEC 8824-1. For readability and for easy referencing the value assignment is given below. The ISO/IEC 8824-1 supersedes this informative note in case of discrepancy.

```
!"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
```

Standard Character Set	Code Page (Decimal)
ISO/IEC 8859-1 Latin-1 Western European	1
ISO/IEC 8859-2 Latin-2 Central European	2
ISO/IEC 8859-3 Latin-3 South European	3
ISO/IEC 8859-5 Latin / Cyrillic	5
ISO/IEC 8859-7 Latin / Greek	7
ISO/IEC 8859-9 Latin-5 Turkish	9
ISO/IEC 8859-13 Latin-7 Baltic Rim	13
ISO/IEC 8859-15 Latin-9	15
ISO/IEC 8859-16 Latin-10 South Eastern European	16
KOI8-R Latin / Cyrillic	80
KOI8-U Latin / Cyrillic	85

5. Encoding

When encoded with ASN.1 encoding rules, all data types defined shall be encoded according to ISO/IEC 8825-2, aligned variant.

6. Object Identifiers and Application Identifiers

6.1. Object Identifiers

The Object Identifiers (OIDs) listed in this chapter are only relevant for generation 2. These OIDs are specified in TR-03110-3 and repeated here for the sake of completeness. These OIDs are contained in the subtree of bsi-de:

```
bsi-de OBJECT IDENTIFIER ::= {
itu-t(0) identified-organization(4) etsi(0)
reserved(127) etsi-identified-organization(0) 7
}
```

VU Authentication protocol identifiers

```
id-TA          OBJECT IDENTIFIER ::= { bsi-de protocols(2) smartcard(2) 2 }
id-TA-ECDSA    OBJECT IDENTIFIER ::= { id-TA 2 }
id-TA-ECDSA-SHA-256 OBJECT IDENTIFIER ::= { id-TA-ECDSA 3 }
id-TA-ECDSA-SHA-384 OBJECT IDENTIFIER ::= { id-TA-ECDSA 4 }
id-TA-ECDSA-SHA-512 OBJECT IDENTIFIER ::= { id-TA-ECDSA 5 }
```

Example: Suppose VU Authentication is to be done with SHA-384, then the object identifier to use is (in ASN.1 notation) bsi-de protocols(2) smartcard(2) 2 2 4. The value of this object identifier in dot notation is 0.4.0.127.0.7.2.2.2.4.

	Dot notation	Byte notation
id-TA-ECDSA-SHA-256	0.4.0.127.0.7.2.2.2.2.3	'04 00 7F 00 07 02 02 02 02 03'
id-TA-ECDSA-SHA-384	0.4.0.127.0.7.2.2.2.2.4	'04 00 7F 00 07 02 02 02 02 04'
id-TA-ECDSA-SHA-512	0.4.0.127.0.7.2.2.2.2.5	'04 00 7F 00 07 02 02 02 02 05'

Chip Authentication protocol identifiers

id-CA OBJECT IDENTIFIER ::= {bsi-de protocols(2) smartcard(2) 3}

id-CA-ECDH OBJECT IDENTIFIER ::= {id-CA 2}

id-CA-ECDH-AES-CBC-CMAC-128 OBJECT IDENTIFIER ::= {id-CA-ECDH 2}

id-CA-ECDH-AES-CBC-CMAC-192 OBJECT IDENTIFIER ::= {id-CA-ECDH 3}

id-CA-ECDH-AES-CBC-CMAC-256 OBJECT IDENTIFIER ::= {id-CA-ECDH 4}

Example: Suppose Chip Authentication is to be done by using the ECDH algorithm, resulting in an AES session key length of 128 bits. This session key will subsequently be used in the CBC mode of operation to ensure data confidentiality and with the CMAC algorithm to ensure data authenticity. Therefore, the object identifier to use is (in ASN.1 notation) bsi-de protocols(2) smartcard(2) 3 2 2. The value of this object identifier in dot notation is 0.4.0.127.0.7.2.2.3.2.2.

	Dot notation	Byte notation
id-CA-ECDH-AES-CBC-CMAC-128	0.4.0.127.0.7.2.2.3.2.2	'04 00 7F 00 07 02 02 03 02 02'
id-CA-ECDH-AES-CBC-CMAC-192	0.4.0.127.0.7.2.2.3.2.3	'04 00 7F 00 07 02 02 03 02 03'
id-CA-ECDH-AES-CBC-CMAC-256	0.4.0.127.0.7.2.2.3.2.4	'04 00 7F 00 07 02 02 03 02 04'

6.2. Application identifiers

Generation 2:

The Application Identifier (AID) for the External GNSS Facility (Generation 2) is given by 'FF 44 54 45 47 4D'. This is a proprietary AID according to ISO/IEC 7816-4.

Note: The last 5 bytes encode DTEGM for smart Tachograph External GNSS Facility.

The Application Identifier for the generation 2 tachograph card application is given by 'FF 53 4D 52 44 54'. This is a proprietary AID according to ISO/IEC 7816-4.

Appendix Sub-appendix 2. Tachograph Cards Specification

TABLE OF CONTENT

1.	INTRODUCTION	234
1.1.	<i>Abbreviations</i>	234
1.2.	<i>References.....</i>	235
2.	ELECTRICAL AND PHYSICAL CHARACTERISTICS	235
2.1.	<i>Supply Voltage and Current Consumption</i>	236
2.2.	<i>Programming Voltage V_{pp}.....</i>	236
2.3.	<i>Clock generation and Frequency.....</i>	236
2.4.	<i>I/O Contact.....</i>	236
2.5.	<i>States of the Card</i>	236
3.	HARDWARE AND COMMUNICATION	237
3.1.	<i>Introduction</i>	237
3.2.	<i>Transmission Protocol.....</i>	237
3.2.1	<i>Protocols</i>	237
3.2.2	<i>ATR.....</i>	238
3.2.3	<i>PTS</i>	238
3.3.	<i>Access Rules</i>	239
3.4.	<i>Commands and error codes overview.....</i>	241
3.5.	<i>Command descriptions</i>	244
3.5.1	<i>SELECT.....</i>	244
3.5.2	<i>Read Binary.....</i>	244
3.5.3	<i>Update Binary.....</i>	246
3.5.4	<i>Get Challenge.....</i>	252
3.5.5	<i>Verify.....</i>	257
3.5.6	<i>Get Response</i>	258
3.5.7	<i>PSO: Verify Certificate.....</i>	259
3.5.8	<i>Internal Authenticate.....</i>	260
3.5.9	<i>External Authenticate</i>	262
3.5.10	<i>General Authenticate.....</i>	263
3.5.11	<i>Manage Security Environment.....</i>	265
3.5.12	<i>PSO: Hash.....</i>	265
3.5.13	<i>Perform Hash of File.....</i>	268
3.5.14	<i>PSO: Compute Digital Signature</i>	269
3.5.15	<i>PSO: Verify Digital Signature.....</i>	271
3.5.16	<i>Process DSRC Message</i>	272
4.	TACHOGRAPH CARDS STRUCTURE.....	273
4.1.	<i>Master File MF.....</i>	275
4.2.	<i>Driver card applications.....</i>	275
4.2.1	<i>Driver card application generation 1</i>	277

4.2.2	<i>Driver card application generation 2</i>	280
4.3.	<i>Workshop card applications</i>	286
4.3.1	<i>Workshop card application generation 1</i>	286
4.3.2	<i>Workshop card application generation 2</i>	289
4.4.	<i>Control card applications</i>	299
4.4.1	<i>Control Card application generation 1</i>	299
4.4.2	<i>Control card application generation 2</i>	301
4.5.	<i>Company card applications</i>	304
4.5.1	<i>Company card application generation 1</i>	304
4.5.2	<i>Company card application generation 2</i>	305

1. Introduction

1.1. Abbreviations

For the purpose of this **sub**-appendix, the following abbreviations apply.

AC	Access conditions
AES	Advanced Encryption Standard
AID	Application Identifier
ALW	Always
APDU	Application Protocol Data Unit (structure of a command)
ATR	Answer To Reset
AUT	Authenticated.
C6, C7	Contacts N° 6 and 7 of the card as described in ISO/IEC 7816-2
cc	clock cycles
CHA	Certificate Holder Authorisation
CHV	Card holder Verification Information
CLA	Class byte of an APDU command
DSRC	Dedicated Short Range Communication
DF	Dedicated File. A DF can contain other files (EF or DF)
DO	Data Object
ECC	Elliptic Curve Cryptography
EF	Elementary File
etu	elementary time unit
G1	Generation 1
G2	Generation 2
IC	Integrated Circuit
ICC	Integrated Circuit Card
ID	Identifier
IFD	Interface Device
IFS	Information Field Size
IFSC	Information Field Size for the card
IFSD	Information Field Size Device (for the Terminal)
INS	Instruction byte of an APDU command
Lc	Length of the input data for a APDU command
Le	Length of the expected data (output data for a command)
MF	Master File (root DF)
NAD	Node Address used in T=1 protocol

NEV	Never
P1-P2	Parameter bytes
PIN	Personal Identification Number
PRO SM	Protected with secure messaging
PTS	Protocol Transmission Selection
RFU	Reserved for Future Use
RST	Reset (of the card)
SFID	Short EF Identifier
SM	Secure Messaging
SW1-SW2	Status bytes
TS	Initial ATR character
VPP	Programming Voltage
VU	Vehicle Unit
XXh	Value XX in hexadecimal notation
' XXh '	Value XX in hexadecimal notation
	Concatenation symbol 03 04=0304

1.2. References

The following references are used in this ~~Appendix~~ **Sub-appendix**:

ISO/IEC 7816-2	Identification cards - Integrated circuit cards - Part 2: Dimensions and location of the contacts. ISO/IEC 7816-2:2007.
ISO/IEC 7816-3	Identification cards - Integrated circuit cards - Part 3: Electrical interface and transmission protocols. ISO/IEC 7816-3:2006.
ISO/IEC 7816-4	Identification cards - Integrated circuit cards - Part 4: Organization, security and commands for interchange. ISO/IEC 7816-4:2013 + Cor 1: 2014.
ISO/IEC 7816-6	Identification cards - Integrated circuit cards - Part 6: Interindustry data elements for interchange. ISO/IEC 7816-6:2004 + Cor 1: 2006.
ISO/IEC 7816-8	Identification cards - Integrated circuit cards - Part 8: Commands for security operations. ISO/IEC 7816-8:2004.
ISO/IEC 9797-2	Information technology -- Security techniques -- Message Authentication Codes (MACs) -- Part 2: Mechanisms using a dedicated hash-function. ISO/IEC 9797-2:2011

2. Electrical and physical characteristics

TCS_01	All electronic signals shall be in accordance with ISO/IEC 7816-3 unless specified otherwise.
TCS_02	The location and dimensions of the card contacts shall comply with the ISO/IEC 7816-2.

2.1. Supply Voltage and Current Consumption

- TCS_03** The card shall work according to specifications within the consumption limits specified in ISO/IEC 7816-3.
- TCS_04** The card shall work with $V_{cc} = 3V (\pm 0.3V)$ or with $V_{cc} = 5V (\pm 0.5V)$. Voltage selection shall be performed according to ISO/IEC 7816-3.

2.2. Programming Voltage V_{pp}

- TCS_05** The card shall not require a programming voltage at pin C6. It is expected that pin C6 is not connected in an IFD. Contact C6 may be connected to V_{cc} in the card but shall not be connected to ground. This voltage should not be interpreted in any case.

2.3. Clock generation and Frequency

- TCS_06** The card shall operate within a frequency range of 1 to 5 MHz and may support higher frequencies. Within one card session the clock frequency may vary $\pm 2\%$. The clock frequency is generated by the Vehicle Unit and not the card itself. The duty cycle may vary between 40 and 60%.
- TCS_07** Under conditions contained into the card file EF ICC, the external clock can be stopped. The first byte of the EF ICC file body codes the Clockstop mode conditions:

<i>Low</i>		<i>High</i>		
Bit 3	Bit 2	Bit 1		
0	0	1		Clockstop allowed, no preferred level
0	1	1		Clockstop allowed, high level preferred
1	0	1		Clockstop allowed, low level preferred
0	0	0		Clockstop not allowed
0	1	0		Clockstop only allowed on high level
1	0	0		Clockstop only allowed on low level

Bits 4 to 8 are not used.

2.4. I/O Contact

- TCS_08** The I/O contact C7 is used to receive data from and to transmit data to the IFD. During operation only either the card or the IFD shall be in transmit mode. Should both units be in transmit mode no damage shall occur to the card. Unless transmitting, the card shall enter the reception mode.

2.5. States of the Card

- TCS_09** The card works in two states while the supply voltage is applied:
- Operation state while executing commands or interfacing with ~~Digital~~ **Vehicle Unit**,
- Idle state at all other times; in this state all data shall be retained by the card.

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

TCS_10 The Transmission protocol shall be compliant with ISO/IEC 7816-3 for T = 0 and T = 1. In particular, the VU shall recognise waiting time extensions sent by the card.

3.2.1 Protocols

TCS_11 The card shall provide both protocol **T=0** and protocol **T=1**. In addition the card may support further contact-oriented protocols.

TCS_12 **T=0** is the default protocol, a **PTS** command is therefore necessary to change the protocol to **T=1**.

TCS_13 Devices shall support direct convention in both protocols: the direct convention is hence mandatory for the card.

TCS_14 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:

TCS_15 **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.

TCS_16 **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

TCS_17 The device checks ATR bytes, according to ISO/IEC 7816-3. No verification shall be done on ATR Hist

Historical Characters.

Example of Basic Biprotocol ATR according to ISO/IEC 7816-3

<i>Character</i>	<i>Value</i>	<i>Remarks</i>
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' (at least 'F0h')	Information Field Size Card (IFSC)
TH1 to TH5	'XXh'	Historical characters
TCK	'XXh'	Check Character (exclusive OR)

TCS_18 After the Answer To Reset (ATR), the Master File (MF) is implicitly selected and becomes the Current Directory.

3.2.3 PTS

TCS_19 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.

TCS_20 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

TCS_21 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:

<i>Character</i>	<i>Value</i>	<i>Remarks</i>
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 rules

TCS_22 An access rule specifies for an access mode, i.e. command, the corresponding security conditions. If these security conditions are fulfilled the corresponding command is processed.

TCS_23 The following security conditions are used for the tachograph card:

<i>Abbreviation</i>	<i>Meaning</i>
ALW	The action is always possible and can be executed without any restriction. Command and response APDU are sent in plain text, i.e. without secure messaging.
NEV	The action is never possible.
PLAIN-C	The command APDU is sent in plain, i.e. without secure messaging.
PWD	The action may only be executed if the workshop card PIN has been successfully verified, i.e. if the card internal security status "PIN_Verified" is set. The command must be sent without secure messaging.
EXT-AUT-G1	The action may only be executed if the External Authenticate command for the generation 1 authentication (see also Appendix Sub-appendix 11 Part A) has been successfully performed.
SM-MAC-G1	The APDU (command and response) must be applied with generation 1 secure messaging in authentication-only mode (see Appendix Sub-appendix 11 Part A).
SM-C-MAC-G1	The command APDU must be applied with generation 1 secure messaging in authentication only mode (see Appendix Sub-appendix 11 Part A).
SM-R-ENC-G1	The response APDU must be applied with generation 1 secure messaging in encryption mode (see Appendix Sub-appendix 11 Part A), i.e. no message authentication code is returned.
SM-R-ENC-MAC-G1	The response APDU must be applied with generation 1 secure messaging in encrypt-then-authenticate mode (see Appendix Sub-appendix 11 Part A).
SM-MAC-G2	The APDU (command and response) must be applied with generation 2 secure messaging in authentication-only mode (see Appendix Sub-appendix 11 Part B).
SM-C-MAC-G2	The command APDU must be applied with generation 2 secure messaging in authentication only mode (see Sub-appendix 11 Part B).

Abbreviation	Meaning
SM-R-ENC-MAC-G2	The response APDU must be applied with generation 2 secure messaging in encrypt-then-authenticate mode (see Appendix Sub-appendix 11 Part B).

TCS_24 These security conditions can be linked in the following ways:

AND: All security conditions must be fulfilled

OR: At least one security condition must be fulfilled

The access rules for the file system, i.e. the SELECT, READ BINARY and UPDATE BINARY command, are specified in chapter □. The access rules for the remaining commands are specified in the following tables. **The term ‘not applicable’ is used if there is no requirement to support the command. In this case the command may or may not be supported, but the access condition is out of scope.**

TCS_25 In the DF Tachograph G1 application the following access rules are used:

Command	Driver Card	Workshop Card	Control Card	Company Card
External Authenticate				
• For generation 1 authentication	ALW	ALW	ALW	ALW
• For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate	ALW	PWD	ALW	ALW
General Authenticate	ALW	ALW	ALW	ALW
Get Challenge	ALW	ALW	ALW	ALW
MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Compute Digital Signature	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Hash	Not applicable	Not applicable	ALW	Not applicable
PERFORM HASH of FILE	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW
PSO: Verify Digital Signature	Not applicable	Not applicable	ALW	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable

TCS_26 In the DF Tachograph_G2 application the following access rules are used:

Command	Driver Card	Workshop Card	Control Card	Company Card
External Authenticate				
- For generation 1 authentication	Not applicable	Not applicable	Not applicable	Not applicable
- For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate	Not applicable	Not applicable	Not applicable	Not applicable
General Authenticate	ALW	ALW	ALW	ALW
Get Challenge	ALW	ALW	ALW	ALW

MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	ALW	ALW	Not applicable
PSO: Compute Digital Signature	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Hash	Not applicable	Not applicable	ALW	Not applicable
PERFORM HASH of FILE	ALW OR SM-MAC-G2	ALW OR SM-MAC-G2	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW
PSO: Verify Digital Signature	Not applicable	Not applicable	ALW	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable

TCS_27 In the MF the following access rules are used:

<i>Command</i>	<i>Driver Card</i>	<i>Workshop Card</i>	<i>Control Card</i>	<i>Company Card</i>
External Authenticate				
- For generation 1 authentication	Not applicable	Not applicable	Not applicable	Not applicable
- For generation 2 authentication	ALW	PWD	ALW	ALW
Internal Authenticate				
General Authenticate	Not applicable	Not applicable	Not applicable	Not applicable
Get Challenge	ALW	ALW	ALW	ALW
MSE:SET AT	ALW	ALW	ALW	ALW
MSE:SET DST	ALW	ALW	ALW	ALW
Process DSRC Message	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Compute Digital Signature	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Hash	Not applicable	Not applicable	Not applicable	Not applicable
PERFORM HASH of FILESO: Hash of File	Not applicable	Not applicable	Not applicable	Not applicable
PSO: Verify Certificate	ALW	ALW	ALW	ALW
PSO: Verify Digital Signature	Not applicable	Not applicable	Not applicable	Not applicable
Verify	Not applicable	ALW	Not applicable	Not applicable

TCS_28 A tachograph card may or may not accept a command with a higher level of security than the one specified in the security conditions. I.e. if the security condition is ALW (or PLAIN-C) the card may accept a command with secure messaging (encryption and / or authentication mode). If the security condition requires secure messaging with authentication mode, the tachograph card may accept a command with secure messaging of the same generation in authentication and encryption mode.

Note: The command descriptions provide more information on the support of the commands for the different tachograph card types and the different DFs.

3.4. 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. The command variants which are supported by a generation 1 and 2 application are specified in the corresponding command descriptions.

<i>Command</i>	<i>INS</i>
SELECT	'A4h'
READ BINARY	'B0h', 'B1h'
UPDATE BINARY	'D6h', 'D7h'
GET CHALLENGE	'84h'
VERIFY	'20h'
GET RESPONSE	'C0h'
PERFORM SECURITY OPERATION	'2Ah'
- VERIFY CERTIFICATE	
- COMPUTE DIGITAL SIGNATURE	
- VERIFY DIGITAL SIGNATURE	
- HASH	
- PERFORM HASH OF FILE	
- PROCESS DSRC MESSAGE	
INTERNAL AUTHENTICATE	'88h'
EXTERNAL AUTHENTICATE	'82h'
MANAGE SECURITY ENVIRONMENT	'22h'
• SET DIGITAL SIGNATURE TEMPLATE	
• SET AUTHENTICATION TEMPLATE	
GENERAL AUTHENTICATE	'86h'

TCS_29 The status words SW1 SW2 are returned in any response message and denote the processing state of the command.

<i>SW1</i>	<i>SW2</i>	<i>Meaning</i>
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	00	Authentication failed (Warning)
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

<i>SW1</i>	<i>SW2</i>	<i>Meaning</i>
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) 68 82 Secure messaging not supported
68	83	Last command of the chain expected
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	80	Incorrect parameters in data field
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

Additional status words as defined in ISO/IEC 7816-4 can be returned, if their behaviour is not explicitly mentioned in this sub-appendix.

For example, the following status words can be optionally returned:

6881: Logical channel not supported

6882: Secure messaging not supported

TCS_30 If more than one error condition is fulfilled in one command APDU the card may return any of the appropriate status words.

3.5. Command descriptions

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

Additional relevant details, related to cryptographic operations involved, are given in ~~Appendix~~ **Sub-appendix 11** Common security mechanisms for Tachograph Generation 1 and Generation 2.

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.

TCS_31 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 ~~§Error! Reference source not found~~ **3.5.6**) command with P3=Le.

TCS_32 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. **TCS_33**

TCS_33 A tachograph card may support extended length fields according to ISO/IEC 7816-4 as an optional feature. A tachograph card that supports extended length fields shall

- Indicate the extended length field support in the ATR
- Provide the supported buffer sizes by means of the extended length information in the EF ATR/INFO see **TCS_146**.
- Indicate whether it supports extended length fields for T = 1 and / or T = 0 in the EF Extended Length, see **TCS_147**.
- Support extended length fields for the tachograph application generation 1 and 2.

Notes:

All commands are specified for short length fields. The usage of extended length APDUs is clear from ISO/IEC 7816-4.

In general the commands are specified for the plain mode, i.e. without secure messaging, as the secure messaging layer is specified in ~~Appendix~~ **Sub-appendix 11**. It is clear from the access rules for a command whether the command shall support secure messaging or not and whether the command shall support generation 1 and / or generation 2 secure messaging. Some command variants are described with secure messaging to illustrate the usage of secure messaging.

TCS_34 The VU shall perform the complete generation 2 VU – card mutual authentication protocol for a session including the certificate verification (if required) either in the DF Tachograph, the DF Tachograph_G2 or the MF.

3.5.1 SELECT

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

The SELECT 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

3.5.1.1 Selection by name (AID)

This command allows selecting an application DF in the card.

TCS_35 This command can be performed from anywhere in the file structure (after the ATR or at any time).

TCS_36 The selection of an application resets the current security environment. After performing the application selection, no current public key is selected anymore. The EXT-AUT-G1 access condition is also lost. If the command was performed without secure messaging, the former secure messaging session keys are no longer available.

TCS_37 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 Generation 1 tachograph application AID : 'FF 53 4D 52 44 54' for the Generation 2 tachograph application

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

TCS_38 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 command, the state returned is '6900'.
- ◆ If the selected application is considered to be corrupted (integrity error is detected within the file attributes), the processing state returned is '6400' or ~~6581~~'6500'.

3.5.1.2 Selection of an Elementary File using its File Identifier

TCS_39 Command Message

TCS_40 A tachograph card shall support the generation 2 secure messaging as specified in **Sub-appendix 11** Part B for this command variant.

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 command is needed (Le absent in T=1, or no response asked in T=0).

TCS_41 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 command, the state returned is '6900'.
- ◆ If the selected file is considered to be corrupted (integrity error is detected within the file attributes), the processing state returned is '6400' or ~~6584~~ '6500'.

3.5.2 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.

3.5.2.1 Command with offset in P1-P2

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

Note: This command without secure messaging can only be used to read a file that supports the ALW security condition for the Read access mode.

TCS_42 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'B0h'	Read Binary
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

Byte	Length	Value	Description
Le	1	'XXh'	Length of data expected. Number of Bytes to be read.

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

TCS_43 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 security conditions 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 6581'6500'.
- ◆ **If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is '6281'.**

3.5.2.1.1 Command with secure messaging (examples)

This command enables the IFD 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 if the security condition SM-R-ENC-MAC-G1 (generation 1) or SM-R-ENC-MAC-G2 (generation 2) is applied.

TCS_44 Command Message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'B0h'	Read Binary
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	'XXh'	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	'XXh'	L _{CC} : Length of following cryptographic checksum '04h' for Generation 1 secure messaging (see Appendix Sub-appendix 11 Part A)

‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see **Appendix Sub-appendix 11 Part B**)

#11-#(10+L)	L	‘XX..XXh’	Cryptographic checksum
Le	1	‘00h’	As specified in ISO/IEC 7816-4

TCS_45 Response Message if SM-R-ENC-MAC-G1 (generation 1) / SM-R-ENC-MAC-G2 (generation 2) is not required 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	‘99h’	Tag for Processing Status (SW1-SW2) – optional for generation 1 secure messaging
#(3+L+NN)	1	‘02h’	Length of Processing Status – optional for generation 1 secure messaging
#(4+L+NN) - #(5+L+NN)	2	‘XX XXh’	Processing Status of the unprotected response APDU – optional for generation 1 secure messaging
#(6+L+NN)	1	‘8Eh’	TCC : Tag for cryptographic checksum
#(7+L+NN)	1	‘XXh’	LCC : Length of following cryptographic checksum ‘04h’ for Generation 1 secure messaging (see Appendix Sub-appendix 11 Part A) ‘08h’, ‘0Ch’ or ‘10h’ depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(8+L+NN)- #(7+M+L+NN)	M	‘XX..XXh’	Cryptographic checksum
SW	2	‘XXXXh’	Status Words (SW1,SW2)

TCS_46 Response Message if SM-R-ENC-MAC-G1 (generation 1) / SM-R-ENC-MAC-G2 (generation 2) is required 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 bytes if L _{PI CG} > 127 bytes.
#(2+L)-#(1+L+MM)	MM	‘01XX..XXh’	Encrypted Data : Padding Indicator and cryptogram
#(2+L+MM)	1	‘99h’	Tag for Processing Status (SW1-SW2) – optional for generation 1 secure messaging
#(3+L+MM)	1	‘02h’	Length of Processing Status – optional for generation 1 secure messaging
#(4+L+MM) - #(5+L+MM)	2	‘XX XXh’	Processing Status of the unprotected response APDU – optional for generation 1 secure messaging
#(6+L+MM)	1	‘8Eh’	TCC : Tag for cryptographic checksum
#(7+L+MM)	1	‘XXh’	LCC : Length of following cryptographic checksum ‘04h’ for Generation 1 secure messaging (see Appendix Sub-appendix 11 Part A)

			'08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(8+L+MM)-#(7+N+L+MM)	N	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

- The READ BINARY command may return regular processing states listed in **TCS_43** under Tag '99h' as described in **TCS_59** using the secure messaging response structure.
- **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:**

TCS_47 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**'.

3.5.2.2 Command with short EF (Elementary File) identifier

This command variant enables the IFD to select an EF by means of a short EF identifier and read data from this EF.

TCS_48 A tachograph card shall support this command variant for all Elementary Files with a specified short EF identifier. These short EF identifiers are specified in chapter □.

TCS_49 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'B0h'	Read Binary
P1	1	'XXh'	Bit 8 is set to 1 Bit 7 and 6 are set to 00 Bit 5 – 1 encode the short EF identifier of the corresponding EF
P2	1	'XXh'	Encodes an offset from 0 to 255 bytes in the EF referenced by P1
Le	1	'XXh'	Length of data expected. Number of Bytes to be read.

Note: The short EF identifiers used for the Generation 2 tachograph application are specified in chapter 4.

If P1 encodes a short EF identifier and the command is successful, the identified EF becomes the currently selected EF (current EF).

TCS_50		Response Message	
Byte	Length	Value	Description
#1-#L	L	'XX..XXh'	Data read
SW	2	'XXXXh'	Status Words (SW1,SW2)

- ◆ If the command is successful, the card returns '9000'.
- ◆ If the file corresponding with the short EF identifier is not found, the processing state returned is '6A82'.
- ◆ If the security conditions 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' '6500'.
- ◆ **If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is '6281'.**

3.5.2.3 Command with odd instruction byte

This command variant enables the IFD to read data from an EF with 32768 bytes or more.

TCS_51 A tachograph card which supports EFs with 32768 bytes or more shall support this command variant for these EFs. A tachograph card may or may not support this command variant for other EFs with the exception of the EF Sensor_Installation_Data. See **TCS_156** and **TCS_160**.

TCS_52		Command Message	
Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'B1h'	Read Binary
P1	1	'00h'	Current EF
P2	1	'00h'	
Lc	1	'NNh'	Lc Length of offset data object.
#6-#(5+NN)	NN	'XX..XXh'	Offset data object: Tag '54h' Length '01h' or '02h' Value offset
Le	1	'XXh'	As specified in ISO/IEC 7816-4

The IFD shall encode the offset data object's length with a minimum possible number of octets, i.e. using the length byte '01h' the IFD shall encode an offset from 0 to 255 and using the length byte '02h' an offset from '256' up to '65535' bytes.

In case of T = 0 the card assumes the value Le = "00h" if no secure messaging is applied.

In case of T = 1 the processing state returned is “6700” if Le=“01h”.

TCS_53 Response Message

Byte	Length	Value	Description
#1-#L	L	‘XX..XXh’	Data read encapsulated in a discretionary data object with tag ‘53h’.
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 security conditions 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 ‘6500’.
- ◆ **If an integrity error is detected within the stored data, the card shall return the demanded data, and the processing state returned is ‘6281’.**

3.5.2.3.1 Command with secure messaging (example)

The following example illustrates the usage of secure messaging if the security condition SM-MAC-G2 applies.

TCS_54 Command message

Byte	Length	Value	Description
CLA	1	‘0Ch’	Secure Messaging asked
INS	1	‘B1h’	Read Binary
P1	1	‘00h’	Current EF
P2	1	‘00h’	
Lc	1	‘XXh’	Length of the secured data field
#6	1	‘B3h’	Tag for plain value data encoded in BER-TLV
#7	1	‘NNh’	L _{PV} : length of transmitted data
#(8)-#(7+NN)	NN	‘XX..XXh’	Plain Data encoded in BER-TLV, i.e. the offset data object with tag ‘54’
#(8+NN)	1	‘97h’	T _{LE} : Tag for expected length specification.
#(9+NN)	1	‘01h’	L _{LE} : Length of expected length
#(10+NN)	1	‘XXh’	Expected length specification (original Le): Number of bytes to be read
#(11+NN)	1	‘8Eh’	T _{CC} : Tag for cryptographic checksum

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
#(12+NN)	1	'XXh'	L _{CC} : Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(13+NN)- #(12+M+NN)	M	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/IEC 7816-4

TCS_55 Response message if the command is successful

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
#1	1	'B3h'	Plain Data encoded in BER-TLV
#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 encoded in BER-TLV, i.e. data read encapsulated in a discretionary data object with tag '53h'.
#(2+L+NN)	1	'99h'	Processing Status of the unprotected response APDU
#(3+L+NN)	1	'02h'	Length of Processing Status
#(4+L+NN) - #(5+L+NN)	2	'XX XXh'	Processing Status of the unprotected response APDU
#(6+L+NN)	1	'8Eh'	T _{CC} : Tag for cryptographic checksum
#(7+L+NN)	1	'XXh'	L _{CC} : Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(8+L+NN)-#(7+M+L+NN)	M	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

3.5.3 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.

3.5.3.1 Command with offset in P1-P2

This command enables the IFD to write data into the EF currently selected, without the card verifying the integrity of data received.

Note: This command without secure messaging can only be used to update a file that supports the ALW security condition for the Update access mode.

TCS_56 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
CLA	1	'00h'	

Byte	Length	Value	Description
INS	1	'D6h'	Update Binary
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.

TCS_57 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 security conditions 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 (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'.

3.5.3.1.1 Command with secure messaging (examples)

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.

TCS_58 Command Message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'D6h'	Update Binary
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	L _{PV} : length of transmitted data.

Byte	Length	Value	Description
		'81 NNh'	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	'XXh'	L _{CC} : Length of following cryptographic checksum '04h' for Generation 1 secure messaging (see Appendix Sub-appendix 11 Part A) '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(9+L+NN)- #(8+M+L+NN)	M	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/IEC 7816-4

TCS_59 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'	Processing Status of the unprotected response APDU
#5	1	'8Eh'	T _{CC} : Tag for cryptographic checksum
#6	1	'XXh'	L _{CC} : Length of following cryptographic checksum '04h' for Generation 1 secure messaging (see Appendix Sub-appendix 11 Part A) '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#7-#(6+L)	L	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

The "regular" processing states, described for the UPDATE BINARY command with no secure messaging (see §3.5.3.1), 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:

TCS_60 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'.

3.5.3.2 Command with short EF identifier

This command variant enables the IFD to select an EF by means of a short EF identifier and write data from this EF.

TCS_61 A tachograph card shall support this command variant for all Elementary Files with a specified short EF identifier. These short EF identifiers are specified in chapter 4.

TCS_62 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'D6h'	Update Binary
P1	1	'XXh'	Bit 8 is set to 1 Bit 7 and 6 are set to 00 Bit 5 – 1 encode the short EF identifier of the corresponding EF
P2	1	'XXh'	Encodes an offset from 0 to 255 bytes in the EF referenced by P1
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

TCS_63 Response Message

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

Note: The short EF identifiers used for the generation 2 tachograph application are specified in chapter 4.

If P1 encodes a short EF identifier and the command is successful, the identified EF becomes the currently selected EF (current EF).

- ◆ If the command is successful, the card returns '9000'.
- ◆ If the file corresponding with the short EF identifier is not found, the processing state returned is '6A82'.
- ◆ If the security conditions 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 (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 '6581'.
- ◆ If writing is unsuccessful, the processing state returned is '6581'.

3.5.3.3 Command with odd instruction byte

This command variant enables the IFD to write data to an EF with 32768 bytes or more.

TCS_64 A tachograph card which supports EFs with 32768 bytes or more shall support this command variant for these EFs. A tachograph card may or may not support this command variant for other EFs.

TCS_65 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
CLA	1	'00h'	
INS	1	'D7h'	Update Binary
P1	1	'00h'	Current EF
P2	1	'00h'	
Lc	1	'NNh'	Lc Length of data in the command data field
#6-#(5+NN)	NN	'XX..XXh'	Offset data object with tag '54h' Discretionary data object with tag '53h' that encapsulates the data to be written

The IFD shall encode the offset data object's and the discretionary data object's length with the minimum possible number of octets, i.e. using the length byte '01h' the IFD shall encode an offset/length from 0 to 255 and using the length byte '02h' an offset / length from '256' up to '65535' bytes.

TCS_66 Response Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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 security conditions 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 (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**'.

3.5.3.3.1 Command with secure messaging (example)

The following example illustrates the usage of secure messaging if the security condition SM-MAC-G2 applies.

TCS_67 Command message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
CLA	1	'0Ch'	Secure Messaging asked
INS	1	'D7h'	Update Binary
P1	1	'00h'	Current EF

Byte	Length	Value	Description
P2	1	'00h'	
Lc	1	'XXh'	Length of the secured data field
#6	1	'B3h'	Tag for plain value data encoded in BER-TLV
#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 encoded in BER-TLV, i.e. offset data object with tag '54h' Discretionary data object with tag '53h' that encapsulates the data to be written
#(7+L+NN)	1	'8Eh'	T _{CC} : Tag for cryptographic checksum
#(8+L+NN)	1	'XXh'	L _{CC} : Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#(9+L+NN)- #(8+M+L+NN)	M	'XX..XXh'	Cryptographic checksum
Le	1	'00h'	As specified in ISO/IEC 7816-4

TCS_68 Response message if the command is successful

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'	Processing Status of the unprotected response APDU
#5	1	'8Eh'	T _{CC} : Tag for cryptographic checksum
#6	1	'XXh'	L _{CC} : Length of following cryptographic checksum '08h', '0Ch' or '10h' depending on AES key length for Generation 2 secure messaging (see Appendix Sub-appendix 11 Part B)
#7-#(6+L)	L	'XX..XXh'	Cryptographic checksum
SW	2	'XXXXh'	Status Words (SW1,SW2)

3.5.4 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. TCS_69 The Challenge issued by the card is only valid for the next command, which uses a challenge, sent to the card.

TCS_70 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
INS	1	'84h'	INS
P1	1	'00h'	P1
P2	1	'00h'	P2
Le	1	'08h'	Le (Length of Challenge expected).

TCS_71 **Response Message**

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
#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'.

3.5.5 **VERIFY**

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

Only the workshop card is required to support this command.

Other types of tachograph cards may or may not implement this command, but for these cards no reference CHV is personalized. Therefore these cards cannot perform this command successfully. For other types of tachograph cards than workshop cards the behavior, i.e. the error code returned, is out of the scope of this specification, if this command is sent.

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.

TCS_72 The PIN entered by the user must be ASCII encoded and right padded with 'FFh' bytes up to a length of 8 bytes by the IFD, see also the data type WorkshopCardPIN in ~~Appendix~~ **Sub-appendix 1**.

TCS_73 The tachograph applications generation 1 and 2 shall use the same reference CHV.

TCS_74 The tachograph card shall check whether the command is encoded correctly. If the command is not encoded correctly the card shall not compare the CHV values, not decrement the remaining CHV attempt counter and not reset the security status "PIN_Verified", but abort the command. A command is encoded correctly, if the CLA, INS, P1, P2, Lc bytes have the specified values, Le is absent, and the command data field has the correct length.

TCS_75 If the command is successful, the remaining CHV attempt counter is reinitialised. The initial value of the remaining CHV attempt counter is 5. If the command is successful the card shall set the internal security status "PIN_Verified". The card shall reset this security status, if the card is reset or if the CHV code transmitted in the command does not match the stored reference CHV.

Note: Using the same reference CHV and a global security status prevents that a workshop employee must re-enter the PIN after a selection of another tachograph application DF.

TCS_76 An unsuccessful comparison is recorded in the card, i.e. the remaining CHV attempts counter shall be decremented by one, in order to limit the number of further attempts of the use of the reference CHV.

TCS_77		Command Message	
Byte	Length	Value	Description
CLA	1	'00h'	
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

TCS_78		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).
- If the reference CHV is considered corrupted, the processing state returned is '**6400**' or '**6581**'.
- If Lc is different from '08h', the processing state is '**6700**'.

3.5.6 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.

TCS_79		Command Message	
Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'C0h'	
P1	1	'00h'	

Byte	Length	Value	Description
P2	1	'00h'	
Le	1	'XXh'	Number of bytes expected

TCS_80 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**'.

3.5.7 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.

3.5.7.1 Generation 1 Command – Response pair

TCS_81 This command variant is only supported by a generation 1 tachograph application.

TCS_82 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 §3.5.113.5.11) using its key identifier.

TCS_83 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~~ **Contracting Party or the root public key**.

TCS_84 **Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'2Ah'	Perform Security Operation

Byte	Length	Value	Description
P1	1	'00h'	P1
P2	1	'AEh'	P2: non BER-TLV coded data (concatenation of data elements)
Lc	1	'C2h'	Lc: Length of the certificate, 194 bytes
#6-#199	194	'XX..XXh'	Certificate: concatenation of data elements (as described in Appendix Sub-appendix 11)

TCS_85 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 Sub-appendix 11** for G1 and G2.
- ◆ 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'.
- ◆ Generation 1 only: If the selected public key (used to unwrap the certificate) has a CHA.LSB (CertificateHolderAuthorisation.equipmentType) different from '00' (i.e. is ~~not~~ **neither** the one of a ~~Member State or of Europe- Contracting Party~~ **nor the root certificate**), the processing state returned is '6985'.

3.5.7.2 Generation 2 Command – Response pair

Depending on the curve size ECC certificates may be so long that they cannot be transmitted in a single APDU. In this case command chaining according to ISO/IEC 7816-4 must be applied and the certificate transmitted in two consecutive PSO: Verify Certificate APDUs.

The certificate structure and the domain parameters are defined in **Sub-appendix 11**.

TCS_86 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_343.

TCS_87 Command Message

Byte	Length	Value	Description
CLA	1	'X0h'	CLA byte indicating command chaining: '00h' the only or last command of the chain '10h' not the last command of a chain
INS	1	'2Ah'	Perform Security Operation
P1	1	'00h'	
P2	1	'BEh'	Verify self-descriptive certificate
Lc	1	'XXh'	Length of the command data field, see TCS_88 and TCS_89 .
#6-#5+L	L	'XX..XXh'	DER-TLV encoded data: ECC Certificate Body data object as first data object concatenated with the ECC Certificate Signature data object as second

Byte	Length	Value	Description
			data object or a part of this concatenation. The tag '7F21' and the corresponding length shall not be transmitted. The order of these data objects is fixed.
TCS_88			For short length APDUs the following provisions apply: the IFD shall use the minimum number of APDUs required to transmit the command payload and transmit the maximum number of bytes in the first command APDU according to the value of the Information Field Size Card Byte, see TCS_14. If the IFD behaves differently, the behavior of the card is out of scope. However any value of 'Lc' up to 255 bytes must be supported by the card.
TCS_89			For extended length APDUs the following provisions apply: if the certificate does not fit into a single APDU, the card shall support command chaining. The IFD shall use the minimum number of APDUs required to transmit the command payload and transmit the maximum number of bytes in the first command APDU. If chaining is needed, any value of 'Lc' up to the maximum extended length size indicated must be supported by the card. the IFD behaves differently, the behavior of the card is out of scope. Note: According to Appendix Sub-appendix 11 the card stores the certificate or the relevant contents of the certificate and updates its currentAuthenticatedTime.

The response message structure and status words are as defined in **TCS_85**.

TCS_90	In addition to the error codes listed in 0, the card may return the following error codes:
	<ul style="list-style-type: none"> ◆ If the selected public key (used to unwrap the certificate) has a CHA.LSB (CertificateHolderAuthorisation.equipmentType) that is not suitable for the certificate verification according to Appendix Sub-appendix 11, the processing state returned is '6985'. ◆ If the currentAuthenticatedTime of the card is later than the Certificate Expiration Date, the processing state returned is '6985'. ◆ If the last command of the chain is expected, the card returns '6883'. ◆ If incorrect parameters are sent in the command data field, the card returns '6A80' (also used in case the data objects are not sent in the specified order).

3.5.8 INTERNAL AUTHENTICATE

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

TCS_91	All tachograph cards shall support this command in the DF Tachograph generation 1. The command may or may not be accessible in the MF and/or the DF Tachograph_G2. If so, the command shall terminate with a suitable error code as the private key of the card (Card.SK) for the generation 1 authentication protocol is only accessible in the DF_Tachograph generation 1.
--------	--

Using the INTERNAL AUTHENTICATE command, the IFD can authenticate the card. The authentication process is described in ~~Appendix~~ **Sub-appendix** 11. It includes the following statements:

TCS_92	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~~ **Sub-appendix 11**).

TCS_93 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 Sub-appendix 11)
Le	1	'80h'	Length of the data expected from the card

TCS_94 Response Message

Byte	Length	Value	Description
#1-#128	128	'XX..XXh'	Card authentication token (see Appendix Sub-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'.

TCS_95 If the INTERNAL AUTHENTICATE command is successful, the current **generation 1** session key, if existing, is erased and no longer available. In order to have a new **generation 1** session key available, the EXTERNAL AUTHENTICATE command for the generation 1 authentication mechanism must be successfully performed.

Note: for generation 2 session keys see Sub-appendix 11 CSM_193 and CSM_195. If generation 2 session keys are established and the tachograph card receives the plain INTERNAL AUTHENTICATE command APDU, it aborts the generation 2 secure messaging session and destroys the generation 2 session keys.

3.5.9 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~~ **Sub-appendix 11** for Tachograph G1 and G2 (VU authentication).

TCS_96 The command variant for the generation 1 mutual authentication mechanism is only supported by a generation 1 tachograph application.

TCS_97 The command variant for the second generation VU-card mutual authentication can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34. **If this generation 2 EXTERNAL AUTHENTICATE command is successful, the current generation 1 session key, if existing, is erased and no longer available.**

Note: For generation 2 session keys see Sub-appendix 11 CSM_193 and CSM_195. If generation 2 session keys are established and the tachograph card receives the plain EXTERNAL AUTHENTICATE command APDU, it aborts the generation 2 secure messaging session and destroys the generation 2 session keys.

TCS_98 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	CLA
INS	1	'82h'	INS
P1	1	'00h'	Keys and algorithms implicitly known
P2	1	'00h'	
Lc	1	'XXh'	Lc (Length of the data sent to the card)
#6-#(5+L)	L	'XX..XXh'	Generation 1 authentication: Cryptogram (see Appendix Sub-appendix 11 Part A) Generation 2 authentication: Signature generated by the IFD (see Appendix Sub-appendix 11 Part B)

TCS_99 Response Message

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

If the command is successful, the card returns '**9000**'.

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**'.

If the command is not immediately preceded with a GET CHALLENGE command, the processing state returned is '**6985**'.

The Generation 1 Tachograph application may return the following additional error codes:

- If no Public Key is present in the Security Environment, '**6A88**' is returned.
- 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 selected private key is considered corrupted, the processing state returned is '**6400**' or '**6581**'.

The command variant for the Generation 2 authentication may return the following additional error code:

If signature verification failed, the card returns '**6300**'.

3.5.10 GENERAL AUTHENTICATE

This command is used for the generation 2 chip authentication protocol specified in ~~Appendix~~ **Sub-appendix 11** Part B and is compliant with ISO/IEC 7816-4.

TCS_100 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also **TCS_34**.

TCS_101 Command Message

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'86h'	
P1	1	'00h'	Keys and protocol implicitly known
P2	1	'00h'	
Lc	1	'NNh'	Lc: length of subsequent data field
#6-#(5+L)	L	'7Ch' + L _{7C} + '80h' + L ₈₀ + 'XX..XXh'	DER-TLV encoded ephemeral public key value (see Appendix Sub-appendix 11) The VU shall send the data objects in this order
5 +Le+1	1	'00h'	As specified in ISO/IEC 7816-4

TCS_102 Response Message

Byte	Length	Value	Description
#1-#L	L	'7Ch' + L _{7C} + '81h' + '08h' + 'XX..XXh' + '82h' + L ₈₂ + 'XX..XXh'	DER-TLV encoded Dynamic Authentication Data: nonce and authentication token (see Appendix Sub-appendix 11)
SW	2	'XXXXh'	Status Words (SW1,SW2)

- ◆ If the command is successful, the card returns '9000'.
- ◆ The card returns '6A80' to indicate incorrect parameters in data field.
- ◆ The card returns '6982' if the External Authenticate command has not been performed successfully

The response Dynamic Authentication Data object '7Ch'

- must be present if the operation is successful, i.e. the Status Words are '9000',
- must be absent in case of an execution error or checking error, i.e. if the Status Words are in the range '6400' – '6FFF', and
- may be absent in case of a warning, i.e. if the Status Words are in the range '6200' – '63FF'.

3.5.11 MANAGE SECURITY ENVIRONMENT

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

3.5.11.1 Generation 1 Command – Response pair

This command is compliant with ISO/IEC 7816-4. The use of this command is restricted regarding the related standard.

- TCS_103** This command is only supported by a generation 1 tachograph application.
- TCS_104** The key referenced in the MSE data field remains the current public key until the next correct MSE command, a DF is selected or the card is reset.
- TCS_105** If the key referenced is not (already) present into the card, the security environment remains unchanged.

TCS_106 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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	8	'XX..XXh'	Key identifier as specified in Appendix Sub-appendix 11

TCS_107 Response Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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**'.

3.5.11.2 Generation 2 Command – Response pairs

For the Generation 2 authentication the tachograph card supports the following MSE: Set command versions which are compliant with ISO/IEC 7816-4. These command versions are not supported for the Generation 1 authentication.

3.5.11.2.1 MSE:SET AT for Chip Authentication

The following MSE:SET AT command is used to select the parameters for the Chip Authentication that is performed by a subsequent General Authenticate command.

TCS_108 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34.

TCS_109 MSE:SET AT Command Message for Chip Authentication

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
CLA	1	'00h'	

Byte	Length	Value	Description
INS	1	'22h'	
P1	1	'41h'	Set for internal authentication
P2	1	'A4h'	Authentication
Lc	1	'NNh'	Lc : length of subsequent data field
#6-#(5+L)	L	'80h' + '0Ah' + 'XX..XXh'	DER-TLV encoded cryptographic mechanism reference: Object Identifier of Chip Authentication (value only, Tag '06h' is omitted). See Appendix Sub-appendix 1 for the values of object identifiers; the byte notation shall be used. See Appendix Sub-appendix 11 for guidance on how to select one of these object identifiers.

3.5.11.2.2 MSE:SET AT for VU Authentication

The following MSE:SET AT command is used to select the parameters and keys for the VU Authentication that is performed by a subsequent External Authenticate command.

TCS_110 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_34.

TCS_111 MSE:SET AT Command Message for VU Authentication

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'22h'	
P1	1	'81h'	Set for external authentication
P2	1	'A4h'	Authentication
Lc	1	'NNh'	Lc: length of subsequent data field
#6-#(5+L)	L	'80h' + '0Ah' + 'XX..XXh'	DER-TLV encoded cryptographic mechanism reference: Object Identifier of VU Authentication (value only, Tag '06h' is omitted). See Appendix Sub-appendix 1 for the values of object identifiers; the byte notation shall be used. See Appendix Sub-appendix 11 for guidance on how to select one of these object identifiers.
		'83h' + '08h' + 'XX..XXh'	DER-TLV encoded reference of the VU public key by the Certificate Holder Reference mentioned in its certificate.
		'91h' + L ₉₁ + 'XX..XXh'	DER-TLV encoded compressed representation of the ephemeral public key of the VU that will be used during Chip Authentication (see Appendix Sub-appendix 11)

3.5.11.2.3 MSE:SET DST

The following MSE:SET DST command is used to set a public key either

- ◆ for the verification of a signature that is provided in a subsequent PSO: Verify Digital Signature command or

- ◆ for the signature verification of a certificate that is provided in a subsequent PSO: Verify Certificate command

TCS_112 The command can be performed in the MF, DF Tachograph and DF Tachograph_G2, see also TCS_33.

TCS_113 **MSE:SET DST Command Message**

Byte	Length	Value	Description
CLA	1	'00h'	
INS	1	'22h'	
P1	1	'81h'	Set for verification
P2	1	'B6h'	Digital Signature
Lc	1	'NNh'	Lc: length of subsequent data field
#6-#(5+L)	L	'83h' + '08h' + 'XX...XXh'	DER-TLV encoded reference of a public key, i.e. the Certificate Holder Reference in the certificate of the public key (see Appendix Sub-appendix 11)

For all command versions the response message structure and status words are given by:

TCS_114 **Response Message**

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

- ◆ If the command is successful, the card returns '9000'. The protocol has been selected and initialised.
- ◆ '6A80' indicates incorrect parameters in the command data field.
- ◆ '6A88' indicates that referenced data (i.e. a referenced key) is not available.
- ◆ **If the currentAuthenticatedTime of the card is later than the Expiration Date of the selected public key, the processing stage returned is '6A88'.**

Note: In the case of a MSE: SET AT for VU Authentication command, the referenced key is a VU_MA public key. The card shall set the VU_MA public key for use, if available in its memory, which matches the Certificate Holder Reference (CHR) given in the command data field (the card can identify VU_MA public keys by means of the certificate's CHA field). A card shall return "6A 88" to this command in case only the VU_Sign public key or no public key of the Vehicle Unit is available. See the definition of the CHA field in Sub-appendix 11 and of data type equipmentType in Sub-appendix 1

Similarly, in case an MSE: SET DST command referencing an EQT (i.e. a VU or a card) is sent to a control card, according to CSM_234 the referenced key is always an EQT_Sign key that has to be used for the verification of a digital signature. According to Figure 13 in Sub-appendix 11, the control card will always have stored the relevant EQT_Sign public key. In some cases, the control card may have stored the corresponding EQT_MA public key. The control card shall always set the EQT_Sign public key for use when it receives an MSE: SET DST command.

3.5.12 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 temporarily for the subsequent command PSO: Verify Digital Signature

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

Only the control card is required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. The command may or may not be accessible in the MF.

The control card application generation 1 supports only SHA-1.

TCS_115 The temporarily stored hash value shall be deleted if a new hash value is computed by means of the PSO: HASH command, if a DF is selected, and if the tachograph card is reset.

TCS_116 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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	'XXh'	Length Lc of the subsequent data field
#6	1	'90h'	Tag for the hash code
#7	1	'XXh'	Length L of the hash code: '14h' in Generation 1 application (see Appendix Sub-appendix 11 Part A) '20h', '30h' or '40h' in Generation 2 application (see Appendix Sub-appendix 11 Part B)
#8-#(7+L)	L	'XX..XXh'	Hash code

TCS_117 Response Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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' for SHA-1, '20h' for SHA-256, '30h' for SHA-384, '40h' for SHA-512 (Generation 2 application).

3.5.13 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.

Only the driver card and the workshop card are required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. If a company or control card implements this command, the command shall be implemented as specified in this chapter.

The command may or may not be accessible in the MF. If so, the command shall be implemented as specified in this chapter, i.e. shall not allow the calculation of a hash value, but terminate with a suitable error code.

TCS_118 The PERFORM HASH of FILE command is used to hash the data area of the currently selected transparent EF.

TCS_119 A tachograph card shall support this command only for the EFs that are listed in chapter 4 under the DF_Tachograph and DF_Tachograph_G2 with the following exception. A tachograph card shall not support the command for the EF Sensor_Installation_Data of DF Tachograph_G2.

TCS_120 The result of the hash operation is stored temporarily in the card. It can then be used to get a digital signature of the file, using the PSO: COMPUTE DIGITAL SIGNATURE command.

TCS_121 The temporarily stored hash of file value shall be deleted if a new hash of file value is computed by means of the PSO:PERFORM HASH of FILE command, if a DF is selected, and if the tachograph card is reset.

TCS_122 The Tachograph Generation 1 application shall support SHA-1.

TCS_123 The Tachograph **Generation 2** application shall support ~~SHA-1 and the~~ **SHA-2 algorithm (SHA-256, SHA-384 and or SHA-512 bits) specified by the cipher suite in Sub-appendix 11 Part B for the card signature key Card Sign.**

TCS_124 Command Message

Byte	Length	Value	Description
CLA	1	'80h'	CLA
INS	1	'2Ah'	Perform Security Operation
P1	1	'90h'	Tag: Hash
P2	1	XXh '00h'	P2: Indicates the algorithm to be used for hashing of the data of the currently selected transparent file:
			Algorithm implicitly unknown
			For the Tachograph Generation 1 application: SHA-1
			For the Tachograph Generation 2 application: SHA-2 algorithm (SHA-256, SHA-384 or SHA-512) defined by the cipher suite in Sub-appendix 11 Part B for the card signature key Card_Sign
			00h for SHA-1
			01h for SHA-256
			02h for SHA-384
			03h for SHA-512

TCS_125 Response Message

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

- If the command is successful, the card returns '9000'.

- If the current EF does not allow this command (EF Sensor_Installation_Data in DF Tachograph_G2), 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 or if there is no current EF, the processing state returned is '6986'

3.5.14 PSO: COMPUTE DIGITAL SIGNATURE

This command is used to compute the digital signature of previously computed hash code (see PERFORM HASH of FILE, see §3.5.13-3.5.13).

Only the driver card and the workshop card are required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement ~~this command, but shall not.~~ **In case of the Generation 2 tachograph application, only the driver card and the workshop card** have a **generation 2** signature key. ~~Therefore these, other cards can not~~ **are not able to successfully** perform the command ~~successfully, but~~ **and** terminate with a suitable error code.

The command may or may not be accessible in the MF. If ~~so~~, the command **is not accessible in the MF, it** shall terminate with a suitable error code.

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

TCS_126 This command shall not compute a digital signature of previously computed hash code with the PSO: HASH command.

TCS_127 The card private key is used to compute the digital signature and is implicitly known by the card.

TCS_128 The Generation 1 tachograph application performs a digital signature using a padding method compliant with PKCS1 (see ~~Appendix~~ **Sub-appendix 11** for details).

TCS_129 The Generation 2 tachograph application computes an elliptic curve based digital signature (see ~~Appendix~~ **Sub-appendix 11** for details).

TCS_130 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
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	'NNh'	Length of the expected signature

TCS_131 Response Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
#1-#L	L	'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’.
- If the hash which was computed in a previous Perform Hash of File command is not available, the processing state returned is ‘6985’.

3.5.15 PSO: VERIFY DIGITAL SIGNATURE

This command is used to verify the digital signature, provided as an input, 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.

Only the control card is required to support this command in the DF Tachograph and DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command. The command may or may not be accessible in the MF.

TCS_132 The VERIFY DIGITAL SIGNATURE command always uses the public key selected by the previous Manage Security Environment MSE: Set DST command and the previous hash code entered by a PSO: HASH command.

TCS_133 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 ‘XXh’	Length Lc of the subsequent data field
#6	1	‘9Eh’	Tag for Digital Signature
#7 or #7-#8	L	81-XXh ‘NNh’ or ‘81 NNh’	Length of digital signature (L is 2 bytes if the digital signature is longer than 127 bytes): 128 bytes coded in accordance with Appendix Sub-appendix 11 Part A for Tachograph Generation 1 application Depending on the selected curve for Tachograph Generation 2 application (see Appendix Sub-appendix 11 Part B)
#(7+L)-#(6+L+NN)	NN	‘XX..XXh’	Digital signature content

TCS_134 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~~ **Sub-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’.
- ◆ **If the selected public key (used to verify the digital signature) has a CHA.LSB (CertificateHolderAuthorisation.equipmentType) that is not suitable for the digital signature verification according to Sub-appendix 11, the processing state returned is ‘6985’.**

3.5.16 PROCESS DSRC MESSAGE

This command is used to verify the integrity and authenticity of the DSRC message and to decipher the data communicated from a VU to a control authority or a workshop over the DSRC link. The card derives the encryption key and the MAC key used to secure the DSRC message as described in ~~Appendix~~ **Sub-appendix 11** Part B chapter 13.

Only the control card and the workshop card are required to support this command in the DF Tachograph_G2.

Other types of tachograph cards may or may not implement this command, but shall not have a DSRC master key. Therefore these cards cannot perform the command successfully, but terminate with a suitable error code.

The command may or may not be accessible in the MF and / or the DF Tachograph. If so, the command shall terminate with a suitable error code.

TCS_135 The DSRC master key is accessible only in the DF Tachograph_G2, i.e. the control and workshop card shall support a successful execution of the command only in the DF Tachograph_G2.

TCS_136 The command shall only decrypt the DSRC data and verify the cryptographic checksum, but not interpret the input data.

TCS_137 The order of the data objects in the command data field is fixed by this specification.

TCS_138 Command Message

<i>Byte</i>	<i>Length</i>	<i>Value</i>	<i>Description</i>
CLA	1	‘80h’	Proprietary CLA
INS	1	‘2Ah’	Perform Security Operation
P1	1	‘80h’	Response data: plain value
P2	1	‘B0h’	Command data: plain value encoded in BER-TLV and including SM DOs
Lc	1	‘NNh’	Length Lc of the subsequent data field
#6-#(5+L)	L	‘87h’ + L ₈₇ + ‘XX..XXh’	DER-TLV encoded padding-content indicator byte followed by encrypted tachograph payload. For the padding-content indicator byte the value ‘00h’ (‘no further indication’)

Byte	Length	Value	Description
			<p>according to ISO/IEC 7816-4:2013 Table 52) shall be used. For the encryption mechanism see Appendix Appendix Sub-appendix 11, Part B chapter 13.</p> <p>Allowed values for the length L_{87} are the multiples of the AES block length plus 1 for the padding-content indicator byte, i.e. from 17 bytes up to and including 193 bytes.</p> <p>Note: See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag '87h'.</p>
		'81h' + '10h'	<p>DER-TLV encoded Control Reference Template for Confidentiality nesting the concatenation of the following data elements (see Appendix 1 DDSRCSecurityData and Appendix Appendix Sub-appendix 11 Part B chapter 13):</p> <p>4 byte time stamp</p> <p>3 byte counter</p> <p>8 byte VU serial number</p> <p>1 byte DSRC master key version</p> <p>Note: See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag '81h'.</p>
		'8Eh' + L_{8E} + 'XX..XXh'	<p>DER-TLV encoded MAC over the DSRC message. For the MAC algorithm and calculation see Appendix Appendix Sub-appendix 11, Part B chapter 13.</p> <p>Note: See ISO/IEC 7816-4:2013 Table 49 for the SM data object with tag '8Eh'.</p>
5+Le+1	1	'00h'	As specified in ISO/IEC 7816-4

TCS_139 **Response Message**

Byte	Length	Value	Description
#1-#L	L	'XX..XXh'	Absent (in case of an error) or deciphered data (padding removed)
SW	2	'XXXXh'	Status Words (SW1,SW2)

- If the command is successful, the card returns '**9000**'.
- '**6A80**' indicates incorrect parameters in the command data field (also used in case the data objects are not sent in the specified order).
- '**6A88**' indicates that referenced data is not available, i.e. the referenced DSRC master key is not available.

- ‘6900’ indicates that the verification of the cryptographic checksum or the decryption of the data failed
- ‘6985’ indicates that the 4-byte time stamp provided in the command data field is earlier than cardValidityBegin or later than cardExpiryDate.

4. Tachograph cards structure

This paragraph specifies the file structures of the Tachograph cards for storage of accessible data.

It does not specify card manufacturer dependent internal structures, such as e.g. file headers, nor storage and handling of data elements needed for internal use only such as EuropeanPublicKey, CardPrivateKey, TDesSessionKey or WorkshopCardPin.

TCS_140 A generation 2 tachograph card shall host the Master File MF and a generation 1 and a generation 2 tachograph application of the same type (e.g. driver card applications).

TCS_141 A tachograph card shall support at least the minimum number of records specified for the corresponding applications and shall not support more records than the maximum number of records specified for the corresponding applications.

The maximum and minimum numbers of records are specified in this chapter for the different applications. **In version 2 of generation 2 driver and workshop cards, the generation 1 application 1 shall support the maximum number of records specified in TCS_150 and TCS_158.**

For the security conditions used in the access rules throughout this chapter please refer to chapter 3.3. In general the access mode “read” denotes the READ BINARY command with even and if supported odd INS byte with the exception of the EF Sensor_Installation_Data on the workshop card, see **TCS_156 and TCS_160**. The access mode “update” denotes the Update Binary command with even and if supported odd INS byte and the access mode “select” the SELECT command.

4.1. Master File MF

TCS_142 After its personalisation, the master file MF shall have the following permanent file structure and file access rules:

Note: .The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.

File	File ID	SFID	Access rules	
			Read / Select	Update
MF	‘3F00h’			
—EF ICC	‘0002h’		ALW	NEV
—EF IC	‘0005h’		ALW	NEV
—EF DIR	‘2F00h’	30	ALW	NEV
—EF ATR/INFO (conditional)	‘2F01h’	29	ALW	NEV
—EF Extended_Length	‘0006h’	28	ALW	NEV
—DF Tachograph	‘0500h’		SC1	
—DF Tachograph_G2			SC1	

The following abbreviation for the security condition is used in this table:

SC1 ALW OR SM-MAC-G2

TCS_143 All EF structures shall be transparent.

TCS_144 The Master File MF shall have the following data structure:

File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
MF	63	184		
EF ICC	25	25		
└ CardIccIdentification	25	25		
└ clockStop	1	1	{00}	
└ cardExtendedSerialNumber	8	8	{00..00}	
└ cardApprovalNumber	8	8	{20..20}	
└ cardPersonaliserID	1	1	{00}	
└ embedderIcAssemblerId	5	5	{00..00}	
└ icIdentifier	2	2	{00 00}	
EF IC	8	8		
└ CardChipIdentification	8	8		
└ icSerialNumber	4	4	{00..00}	
└ icManufacturingReferences	4	4	{00..00}	
EF DIR	20	20		
└ See TCS_145	20	20	{00..00}	
EF ATR/INFO	7	128		
└ See TCS_146	7	128	{00..00}	
EF EXTENDED_LENGTH	3	3		
└ See TCS_147	3	3	{00..00}	
DF Tachograph				
DF Tachograph_G2				

TCS_145 The elementary file EF DIR shall contain the following application related data objects: '61 08 4F 06 FF 54 41 43 48 4F 61 08 4F 06 FF 53 4D 52 44 54'

TCS_146 The elementary file EF ATR/INFO shall be present if the tachograph card indicates in its ATR that it supports extended length fields. In this case the EF ATR/INFO shall contain the extended length information data object (DO'7F66') as specified in ISO/IEC 7816-4:2013 clause 12.7.1.

TCS_147 The elementary file EF Extended_Length shall be present if the tachograph card indicates in its ATR that it supports extended length fields. In this case the EF shall contain the following data object: '02 01 xx' where the value 'xx' indicates whether extended length fields are supported for the T = 1 and / or T = 0 protocol.

The value '01' indicates extended length field support for the T = 1 protocol.

The value '10' indicates extended length field support for the T = 0 protocol.

The value '11' indicates extended length field support for the T = 1 and the T = 0 protocol.

4.2. Driver card applications

4.2.1 Driver card application generation 1

TCS_148 After its personalisation, the driver card application generation 1 shall have the following permanent file structure and file access rules:

File	File ID	Access rules		
		Read	Select	Update
└ DF Tachograph	'0500h'		SC1	
├ EF Application_Identification	'0501h'	SC2	SC1	NEV
├ EF Card_Certificate	'C100h'	SC2	SC1	NEV
├ EF CA_Certificate	'C108h'	SC2	SC1	NEV
├ EF Identification	'0520h'	SC2	SC1	NEV
├ EF Card_Download	'050Eh'	SC2	SC1	SC1
├ EF Driving_Licence_Info	'0521h'	SC2	SC1	NEV
├ EF Events_Data	'0502h'	SC2	SC1	SC3
├ EF Faults_Data	'0503h'	SC2	SC1	SC3
├ EF Driver_Activity_Data	'0504h'	SC2	SC1	SC3
├ EF Vehicles_Used	'0505h'	SC2	SC1	SC3
├ EF Places	'0506h'	SC2	SC1	SC3
├ EF Current_Usage	'0507h'	SC2	SC1	SC3
├ EF Control_Activity_Data	'0508h'	SC2	SC1	SC3
├ EF Specific_Conditions	'0522h'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

SC1 ALW OR SM-MAC-G2

SC2 ALW OR SM-MAC-G1 OR SM-MAC-G2

SC3 SM-MAC-G1 OR SM-MAC-G2

TCS_149 All EF structures shall be transparent.

TCS_150 The driver card application generation 1 shall have the following data structure:

File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
└ DF Tachograph		11378	24926	
├ EF Application_Identification		10	10	
├ └ DriverCardApplicationIdentification		10	10	
├ └ └ typeOfTachographCardId		1	1	{00}
├ └ └ cardStructureVersion		2	2	{00 00}
├ └ └ noOfEventsPerType		1	1	{00}
├ └ └ noOfFaultsPerType		1	1	{00}
├ └ └ activityStructureLength		2	2	{00 00}
├ └ └ noOfCardVehicleRecords		2	2	{00 00}
├ └ └ noOfCardPlaceRecords		1	1	{00}
├ EF Card_Certificate		194	194	
├ └ CardCertificate		194	194	{00..00}
├ EF CA_Certificate		194	194	
├ └ MemberStateCertificate		194	194	{00..00}
├ EF Identification		143	143	
├ └ CardIdentification		65	65	
├ └ └ cardIssuingMemberState		1	1	{00}
├ └ └ cardNumber		16	16	{20..20}

		cardIssuingAuthorityName		36	36	{00,20..20}
		cardIssueDate		4	4	{00..00}
		cardValidityBegin		4	4	{00..00}
		cardExpiryDate		4	4	{00..00}
		DriverCardHolderIdentification		78	78	
			cardHolderName	72	72	
			holderSurname	36	36	{00, 20..20}
			holderFirstNames	36	36	{00, 20..20}
		cardHolderBirthDate		4	4	{00..00}
		cardHolderPreferredLanguage		2	2	{20 20}
EF		Card_Download		4	4	
		LastCardDownload		4	4	{00..00}
EF		Driving_Licence_Info		53	53	
		CardDrivingLicenceInformation		53	53	
		drivingLicenceIssuingAuthority		36	36	{00, 20..20}
		drivingLicenceIssuingNation		1	1	{00}
		drivingLicenceNumber		16	16	{20..20}
EF		Events_Data		864	1728	
		CardEventData		864	1728	
		cardEventRecords	6	144	288	
		CardEventRecord	n ₁	24	24	
		eventType		1	1	{00}
		eventBeginTime		4	4	{00..00}
		eventEndTime		4	4	{00..00}
		eventVehicleRegistration				
		vehicleRegistrationNation		1	1	{00}
		vehicleRegistrationNumber		14	14	{00, 20..20}
EF		Faults_Data		576	1152	
		CardFaultData		576	1152	
		cardFaultRecords	2	288	576	
		CardFaultRecord	n ₂	24	24	
		faultType		1	1	{00}
		faultBeginTime		4	4	{00..00}
		faultEndTime		4	4	{00..00}
		faultVehicleRegistration				
		vehicleRegistrationNation		1	1	{00}
		vehicleRegistrationNumber		14	14	{00, 20..20}
EF		Driver_Activity_Data		5548	13780	
		CardDriverActivity		5548	13780	
		activityPointerOldestDayRecord		2	2	{00 00}
		activityPointerNewestRecord		2	2	{00 00}
		activityDailyRecords	n ₆	5544	13776	{00..00}
EF		Vehicles_Used		2606	6202	
		CardVehiclesUsed		2606	6202	
		vehiclePointerNewestRecord		2	2	{00 00}
		cardVehicleRecords		2604	6200	
		CardVehicleRecord	n ₃	31	31	
		vehicleOdometerBegin		3	3	{00..00}
		vehicleOdometerEnd		3	3	{00..00}
		vehicleFirstUse		4	4	{00..00}
		vehicleLastUse		4	4	{00..00}
		vehicleRegistration				

└─vehicleRegistrationNation	1	1	{00}
└─vehicleRegistrationNumber	14	14	{00, 20..20}
└─vuDataBlockCounter	2	2	{00 00}
EF Places	841	1121	
└─CardPlaceDailyWorkPeriod	841	1121	
└─placePointerNewestRecord	1	1	{00}
└─placeRecords	840	1120	
└─PlaceRecord	n ₄	10	10
└─entryTime	4	4	{00..00}
└─entryTypeDailyWorkPeriod	1	1	{00}
└─dailyWorkPeriodCountry	1	1	{00}
└─dailyWorkPeriodRegion	1	1	{00}
└─vehicleOdometerValue	3	3	{00..00}
EF Current_Usage	19	19	
└─CardCurrentUse	19	19	
└─sessionOpenTime	4	4	{00..00}
└─sessionOpenVehicle			
└─vehicleRegistrationNation	1	1	{00}
└─vehicleRegistrationNumber	14	14	{00, 20..20}
EF Control_Activity_Data	46	46	
└─CardControlActivityDataRecord	46	46	
└─controlType	1	1	{00}
└─controlTime	4	4	{00..00}
└─controlCardNumber			
└─cardType	1	1	{00}
└─cardIssuingMemberState	1	1	{00}
└─cardNumber	16	16	{20..20}
└─controlVehicleRegistration			
└─vehicleRegistrationNation	1	1	{00}
└─vehicleRegistrationNumber	14	14	{00, 20..20}
└─controlDownloadPeriodBegin	4	4	{00..00}
└─controlDownloadPeriodEnd	4	4	{00..00}
EF Specific_Conditions	280	280	
└─SpecificConditionRecord	56	5	5
└─entryTime	4	4	{00..00}
└─SpecificConditionType	1	1	{00}

TCS_151 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the driver card data structure must use for a generation 1 application:

		Min	Max
n ₁	NoOfEventsPerType	6	12
n ₂	NoOfFaultsPerType	12	24
n ₃	NoOfCardVehicleRecords	84	200
n ₄	NoOfCardPlaceRecords	84	112
n ₆	CardActivityLengthRange	5544 bytes (28 days * 93 activity changes)	13776 Bytes (28 days * 240 activity changes)

4.2.2 Driver card application generation 2

TCS_152 After its personalisation, the driver card application generation 2 shall have the following permanent file structure and file access rules.

Notes:-

- The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.
- EF Application_Identification_V2, EF Places_Authentication, EF GNSS_Places_Authentication, EF Border_Crossings, EF Load_Unload_Operations, EF VU_Configuration and EF Load_Type_Entries are only present in version 2 of the generation 2 driver card.
- cardStructureVersion in EF Application_Identification is equal to {01 01} for version 2 of the generation 2 driver card, while it was equal to {01 00} for version 1 of the generation 2 driver card.

File	File-ID	SFID	Access rules	
			Read / Select	Update
└ DF Tachograph_G2			SC+	
└ EF Application_Identification	'0501h'	1	SC+	NEV
└ EF CardMA_Certificate	'C100h'	2	SC+	NEV
└ EF CardSignCertificate	'C101h'	3	SC+	NEV
└ EF CA_Certificate	'C108h'	4	SC+	NEV
└ EF Link_Certificate	'C109h'	5	SC+	NEV
└ EF Identification	'0520h'	6	SC+	NEV
└ EF Card_Download	'050Eh'	7	SC+	SC+
└ EF Driving_Licence_Info	'0521h'	10	SC+	NEV
└ EF Events_Data	'0502h'	12	SC+	SM-MAC-G2
└ EF Faults_Data	'0503h'	13	SC+	SM-MAC-G2
└ EF Driver_Activity_Data	'0504h'	14	SC+	SM-MAC-G2
└ EF Vehicles_Used	'0505h'	15	SC+	SM-MAC-G2
└ EF Places	'0506h'	16	SC+	SM-MAC-G2
└ EF Current_Usage	'0507h'	17	SC+	SM-MAC-G2
└ EF Control_Activity_Data	'0508h'	18	SC+	SM-MAC-G2
└ EF Specific_Conditions	'0522h'	19	SC+	SM-MAC-G2
└ EF VehicleUnits_Used	'0523h'	20	SC+	SM-MAC-G2
└ EF GNSS_Places	'0524h'	21	SC+	SM-MAC-G2

File	File ID	SFID	Access rules	
			Read / Select	Update
└ DF Tachograph_G2			SC1	
└ EF Application_Identification	'0501h'	1	SC1	NEV

EF CardMA_Certificate	\C100h	2	SC1	NEV
EF CardSignCertificate	\C101h	3	SC1	NEV
EF CA_Certificate	\C108h	4	SC1	NEV
EF Link_Certificate	\C109h	5	SC1	NEV
EF Identification	\0520h	6	SC1	NEV
EF Card_Download	\050Eh	7	SC1	SC1
EF Driving_Licence_Info	\0521h	10	SC1	NEV
EF Events_Data	\0502h	12	SC1	SM-MAC-G2
EF Faults_Data	\0503h	13	SC1	SM-MAC-G2
EF Driver_Activity_Data	\0504h	14	SC1	SM-MAC-G2
EF Vehicles_Used	\0505h	15	SC1	SM-MAC-G2
EF Places	\0506h	16	SC1	SM-MAC-G2
EF Current_Usage	\0507h	17	SC1	SM-MAC-G2
EF Control_Activity_Data	\0508h	18	SC1	SM-MAC-G2
EF Specific_Conditions	\0522h	19	SC1	SM-MAC-G2
EF VehicleUnits_Used	\0523h	20	SC1	SM-MAC-G2
EF GNSS_Places	\0524h	21	SC1	SM-MAC-G2
EF Application_Identification_V2	\0525h	22	SC1	NEV
EF Places_Authentication	\0526h	23	SC1	SM-MAC-G2
EF GNSS_Places_Authentication	\0527h	24	SC1	SM-MAC-G2
EF Border_Crossings	\0528h	25	SC1	SM-MAC-G2
EF Load_Unload_Operations	\0529h	26	SC1	SM-MAC-G2
EF Load_Type_Entries	\0530h	27	SC1	SM-MAC-G2
EF Vu_Configuration	\0540h	30	SC5/SC1	SM-MAC-G2

The following abbreviations for the security condition are used in this table:

SC1 ALW OR SM-MAC-G2

SC5 For the Read Binary command with even INS byte: SM-C-MAC-G2 AND SM-R-ENC-MAC-G2

For the Read Binary command with odd INS byte (if supported): NEV

TCS_153 All EF structures shall be transparent.

TCS_154 The driver card application generation 2 shall have the following data structure:

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
DF				
Tachograph_G2		98300	98848	
EF Application_Identification		17	17	
DriverCardApplicationIdentification		17	17	
typeOfTachographCardId		1	1	{00}
cardStructureVersion		2	2	{01 01}
noOfEventsPerType		1	1	{00}
noOfFaultsPerType		1	1	{00}
activityStructureLength		2	2	{00 00}
noOfCardVehicleRecords		2	2	{00 00}
noOfCardPlaceRecords		2	2	{00 00}
noOfGNSSADRecords		2	2	{00 00}
noOfSpecificConditionRecords		2	2	{00 00}
noOfCardVehicleUnitRecords		2	2	{00 00}
EF CardMA_Certificate		204	341	
CardMA_Certificate		204	341	{00..00}
EF CardSignCertificate		204	341	
CardSignCertificate		204	341	{00..00}

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
EF CA_Certificate		204	341	
MemberStateCertificate		204	341	{00..00}
EF Link_Certificate		204	341	
LinkCertificate		204	341	{00..00}
EF Identification		143	143	
CardIdentification		65	65	
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}
cardIssuingAuthorityName		36	36	{00, 20..20}
cardIssueDate		4	4	{00..00}
cardValidityBegin		4	4	{00..00}
cardExpiryDate		4	4	{00..00}
DriverCardHolderIdentification		78	78	
cardHolderName		72	72	
holderSurname		36	36	{00, 20..20}
holderFirstNames		36	36	{00, 20..20}
cardHolderBirthDate		4	4	{00..00}
cardHolderPreferredLanguage		2	2	{20 20}
EF Card_Download		4	4	
LastCardDownload		4	4	{00..00}
EF Driving_Licence_Info		53	53	
CardDrivingLicenceInformation		53	53	
drivingLicenceIssuingAuthority		36	36	{00, 20..20}
drivingLicenceIssuingNation		1	1	{00}
drivingLicenceNumber		16	16	{20..20}
EF Events_Data		3168	3168	
CardEventData		3168	3168	
cardEventRecords	11	288	288	
CardEventRecord	n1	24	24	
eventType		1	1	{00}
eventBeginTime		4	4	{00..00}
eventEndTime		4	4	{00..00}
eventVehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
EF Faults_Data		1152	1152	
CardFaultData		1152	1152	
cardFaultRecords	2	576	576	
CardFaultRecord	n2	24	24	
faultType		1	1	{00}
faultBeginTime		4	4	{00..00}
faultEndTime		4	4	{00..00}
faultVehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
EF Driver_Activity_Data		13780	13780	
CardDriverActivity		13780	13780	
activityPointerOldestDayRecord		2	2	{00 00}
activityPointerNewestRecord		2	2	{00 00}
activityDailyRecords	n6	13776	13776	{00..00}
EF Vehicles_Used		9602	9602	
CardVehiclesUsed		9602	9602	
vehiclePointerNewestRecord		2	2	{00 00}
cardVehicleRecords		9600	9600	

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
cardVehicleRecord	n3	48	48	
vehicleOdometerBegin		3	3	{00..00}
vehicleOdometerEnd		3	3	{00..00}
vehicleFirstUse		4	4	{00..00}
vehicleLastUse		4	4	{00..00}
vehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
vuDataBlockCounter		2	2	{00 00}
vehicleIdentificationNumber		17	17	{20..20}
EF Places		2354	2354	
CardPlaceDailyWorkPeriod		2354	2354	
placePointerNewestRecord		2	2	{00 00}
placeRecords		2352	2352	
PlaceRecord	n4	21	21	
entryTime		4	4	{00..00}
entryTypeDailyWorkPeriod		1	1	{00}
dailyWorkPeriodCountry		1	1	{00}
dailyWorkPeriodRegion		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
entryGNSSPlaceRecord		11	11	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
EF Current_Usage		19	19	
CardCurrentUse		19	19	
sessionOpenTime		4	4	{00..00}
sessionOpenVehicle				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
EF Control_Activity_Data		46	46	
CardControlActivityDataRecord		46	46	
controlType		1	1	{00}
controlTime		4	4	{00..00}
controlCardNumber				
cardType		1	1	{00}
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}
controlVehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
controlDownloadPeriodBegin		4	4	{00..00}
controlDownloadPeriodEnd		4	4	{00..00}
EF Specific_Conditions		562	562	
SpecificConditions		562	562	
conditionPointerNewestRecord		2	2	{00 00}
specificConditionRecords		560	560	
SpecificConditionRecord	n9	5	5	
entryTime		4	4	{00..00}
specificConditionType		1	1	{00}
EF VehicleUnits_Used		2002	2002	
CardVehicleUnitsUsed		2002	2002	
vehicleUnitPointerNewestRecord		2	2	{00 00}
cardVehicleUnitRecords		2000	2000	

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
CardVehicleUnitRecord	n7	10	10	
timeStamp		4	4	{00..00}
manufacturerCode		1	1	{00}
deviceID		1	1	{00}
vuSoftwareVersion		4	4	{00..00}
EF GNSS_Places		6050	6050	
GNSSAccumulatedDriving		6050	6050	
gnssADPointerNewestRecord		2	2	{00 00}
gnssAccumulatedDrivingRecords		6048	6048	
GNSSAccumulatedDrivingRecord	n8	18	18	
timeStamp		4	4	{00..00}
gnssPlaceRecord		14	14	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
vehicleOdometerValue		3	3	{00..00}
EF Application_Identification_V2		10	10	
DriverCardApplicationIdentificationV2		10	10	
lengthOfFollowingData		2	2	{00 00}
noOfBorderCrossingRecords		2	2	{00 00}
noOfLoadUnloadRecords		2	2	{00 00}
noOfLoadTypeEntryRecords		2	2	{00 00}
VuConfigurationLengthRange		2	2	{00 00}
EF Places_Authentication		562	562	
CardPlaceAuthDailyWorkPeriod		562	562	
placeAuthPointerNewestRecord		2	2	{00 00}
placeAuthStatusRecords		560	560	
PlaceAuthStatusRecord	n4	5	5	
entryTime		4	4	{00..00}
authenticationStatus		1	1	{00}
EF GNSS_Places_Authentication		1682	1682	
GNSSAuthAccumulatedDriving		1682	1682	
gnssAuthADPointerNewestRecord		2	2	{00 00}
gnssAuthStatusADRecords		1680	1680	
GNSSAuthStatusADRecord	n8	5	5	
timeStamp		4	4	{00..00}
authenticationStatus		1	1	{00}
EF Border_Crossings		19042	19042	
CardBorderCrossings		19042	19042	
borderCrossingPointerNewestRecord		2	2	{00 00}
cardBorderCrossingRecords		19040	19040	
CardBorderCrossingRecord	n10	17	17	
countryLeft		1	1	{00}
countryEntered		1	1	{00}
gnssPlaceAuthRecord		12	12	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
authenticationStatus		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
EF Load_Unload_Operations		32482	32482	
CardLoadUnloadOperations		32482	32482	
loadUnloadPointerNewestRecord		2	2	{00 00}
cardloadUnloadRecords		32480	32480	

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
CardLoadUnloadRecord	n11	20	20	
timestamp		4	4	{00}
operationType		1	1	{00..00}
gnssPlaceAuthRecord		12	12	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
authenticationStatus		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
EF Load Type Entries		1682	1682	
CardLoadTypeEntries		1682	1682	
loadtypeEntryPointerNewestRecord		2	2	{00 00}
cardLoadTypeEntryRecords		1680	1680	
CardLoadTypeEntryRecord	n12	5	5	
timestamp		4	4	{00..00}
loadTypeEntered		1	1	{00}
EF VU Configuration		3072	3072	
VuConfigurations	n13	3072	3072	

TCS_155 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the driver card data structure must use for a generation 2 application:

		Min	Max
n1	NoOfEventsPerType	612	12
n2	NoOfFaultsPerType	1224	24
n3	NoOfCardVehicleRecords	84200	200
n4	NoOfCardPlaceRecords	84112	112
n6	CardActivityLengthRange	5544-13776 Bytes (28-56 days * 93-117 activity changes)	13776 Bytes (28-56 days * 240-117 activity changes)
n7	NoOfCardVehicleUnitRecords	84200	200
n8	NoOfGNSSCDRecords NoOfGNSSADRecords	252336	336
n9	NoOfSpecificConditionRecords	56112	112
n10	NoOfBorderCrossingRecords	1120	1120
n11	NoOfLoadUnloadRecords	1624	1624
n12	NoOfLoadTypeEntryRecords	336	336
n13	VuConfigurationLengthRange	3072 Bytes	3072 Bytes

File	File ID	Access rules		
		Read	Select	Update
└DF Tachograph	'0500h'		SC1	
└EF Application_Identification	'0501h'	SC2	SC1	NEV
└EF Card_Certificate	'C100h'	SC2	SC1	NEV
└EF CA_Certificate	'C108h'	SC2	SC1	NEV
└EF Identification	'0520h'	SC2	SC1	NEV
└EF Card_Download	'0509h'	SC2	SC1	SC1
└EF Calibration	'050Ah'	SC2	SC1	SC3
└EF Sensor_Installation_Data	'050Bh'	SC4	SC1	NEV
└EF Events_Data	'0502h'	SC2	SC1	SC3
└EF Faults_Data	'0503h'	SC2	SC1	SC3
└EF Driver_Activity_Data	'0504h'	SC2	SC1	SC3
└EF Vehicles_Used	'0505h'	SC2	SC1	SC3
└EF Places	'0506h'	SC2	SC1	SC3
└EF Current_Usage	'0507h'	SC2	SC1	SC3
└EF Control_Activity_Data	'0508h'	SC2	SC1	SC3
└EF Specific_Conditions	'0522h'	SC2	SC1	SC3

4.3. Workshop card applications

4.3.1 Workshop card application generation 1

TCS_156 After its personalisation, the workshop card application generation 1 shall have the following permanent file structure and file access rules:

The following abbreviations for the security conditions are used in this table:

SC1 ALW OR SM-MAC-G2

SC2 ALW OR SM-MAC-G1 OR SM-MAC-G2

SC3 SM-MAC-G1 OR SM-MAC-G2

SC4 For the READ BINARY command with even INS byte:

(SM-C-MAC-G1 AND SM-R-ENC-MAC-G1) OR

(SM-C-MAC-G2 AND SM-R-ENC-MAC-G2)

For the READ BINARY command with odd INS byte (if supported): NEV

TCS_157 All EF structures shall be transparent.

TCS_158 The workshop card application generation 1 shall have the following data structure:

File / Data element	No of Records	Size (Bytes)		Default Values
		Min	Max	
└DF Tachograph		11055	29028	
└EF Application_Identification		11	11	
└└WorkshopCardApplicationIdentification		11	11	
└└└typeOfTachographCardId		1	1	{00}
└└└cardStructureVersion		2	2	{00 00}
└└└noOfEventsPerType		1	1	{00}
└└└noOfFaultsPerType		1	1	{00}
└└└activityStructureLength		2	2	{00 00}
└└└noOfCardVehicleRecords		2	2	{00 00}
└└└noOfCardPlaceRecords		1	1	{00}
└└└noOfCalibrationRecords		1	1	{00}
└EF Card_Certificate		194	194	
└└CardCertificate		194	194	{00..00}
└EF CA_Certificate		194	194	

└ MemberStateCertificate		194	194	{00..00}
EF Identification		211	211	
└ CardIdentification		65	65	
└─ cardIssuingMemberState		1	1	{00}
└─ cardNumber		16	16	{20..20}
└─ cardIssuingAuthorityName		36	36	{00, 20..20}
└─ cardIssueDate		4	4	{00..00}
└─ cardValidityBegin		4	4	{00..00}
└─ cardExpiryDate		4	4	{00..00}
└ WorkshopCardHolderIdentification		146	146	
└─ workshopName		36	36	{00, 20..20}
└─ workshopAddress		36	36	{00, 20..20}
└─ cardHolderName				
└─└ holderSurname		36	36	{00, 20..20}
└─└ holderFirstNames		36	36	{00, 20..20}
└─ cardHolderPreferredLanguage		2	2	{20 20}
EF Card_Download		2	2	
└ NoOfCalibrationsSinceDownload		2	2	{00 00}
EF Calibration		9243	26778	
└ WorkshopCardCalibrationData		9243	26778	
└─ calibrationTotalNumber		2	2	{00 00}
└─ calibrationPointerNewestRecord		1	1	{00}
└─ nRecords		9240	26775	
└─└ WorkshopCardCalibrationRecord	n ₅	105	105	
└─└─ calibrationPurpose		1	1	{00}
└─└─ vehicleIdentificationNumber		17	17	{20..20}
└─└─ vehicleRegistration				
└─└─└ vehicleRegistrationNation		1	1	{00}
└─└─└ vehicleRegistrationNumber		14	14	{00, 20..20}
└─└─ wVehicleCharacteristicConstan		2	2	{00 00}
└─└─ kConstantOfRecordingEquipment		2	2	{00 00}
└─└─ lTyreCircumference		2	2	{00 00}
└─└─ tyreSize		15	15	{20..20}
└─└─ authorisedSpeed		1	1	{00}
└─└─ oldOdometerValue		3	3	{00..00}
└─└─ newOdometerValue		3	3	{00..00}
└─└─ oldTimeValue		4	4	{00..00}
└─└─ newTimeValue		4	4	{00..00}
└─└─ ationDate		4	4	{00..00}
└─└─ vuPartNumber		16	16	{20..20}
└─└─ vuSerialNumber		8	8	{00..00}
└─└─ sensorSerialNumber		8	8	{00..00}
EF Sensor_Installation_Data		16	16	
└ SensorInstallationSecData		16	16	{00..00}
EF Events_Data		432	432	
└ ata		432	432	
└─ cardEventRecords	6	72	72	
└─└ CardEventRecord	n ₁	24	24	
└─└─ eventType		1	1	{00}
└─└─ eventBeginTime		4	4	{00..00}
└─└─ eventEndTime		4	4	{00..00}
└─└─ eventVehicleRegistration				
└─└─└ vehicleRegistrationNation		1	1	{00}

└vehicleRegistrationNumber		14	14	{00, 20..20}
EF Faults_Data		288	288	
└ata		288	288	
└cardFaultRecords	2	144	144	
└CardFaultRecord	n ₂	24	24	
└faultType		1	1	{00}
└faultBeginTime		4	4	{00..00}
└faultEndTime		4	4	{00..00}
└faultVehicleRegistration				
└vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
EF Driver_Activity_Data		202	496	
└CardDriverActivity		202	496	
└activityPointerOldestDayRecord		2	2	{00 00}
└activityPointerNewestRecord		2	2	{00 00}
└activityDailyRecords	n ₆	198	492	{00..00}
EF Vehicles_Used		126	250	
└CardVehiclesUsed		126	250	
└vehiclePointerNewestRecord		2	2	{00 00}
└cardVehicleRecords		124	248	
└CardVehicleRecord	n ₃	31	31	
└vehicleOdometerBegin		3	3	{00..00}
└vehicleOdometerEnd		3	3	{00..00}
└vehicleFirstUse		4	4	{00..00}
└vehicleLastUse		4	4	{00..00}
└vehicleRegistration				
└vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
└vuDataBlockCounter		2	2	{00 00}
EF Places		61	81	
└CardPlaceDailyWorkPeriod		61	81	
└placePointerNewestRecord		1	1	{00}
└ds		60	80	
└PlaceRecord	n ₄	10	10	
└entryTime		4	4	{00..00}
└entryTypeDailyWorkPeriod		1	1	{00}
└dailyWorkPeriodCountry		1	1	{00}
└dailyWorkPeriodRegion		1	1	{00}
└vehicleOdometerValue		3	3	{00..00}
EF Current_Usage		19	19	
└CardCurrentUse		19	19	
└sessionOpenTime		4	4	{00..00}
└sessionOpenVehicle				
└vehicleRegistrationNation		1	1	{00}
└vehicleRegistrationNumber		14	14	{00, 20..20}
EF Control_Activity_Data		46	46	
└CardControlActivityDataRecord		46	46	
└controlType		1	1	{00}
└controlTime		4	4	{00..00}
└controlCardNumber				
└cardType		1	1	{00}
└cardIssuingMemberState		1	1	{00}

└─cardNumber	16	16	{20..20}
└─controlVehicleRegistration			
└─vehicleRegistrationNation	1	1	{00}
└─vehicleRegistrationNumber	14	14	{00, 20..20}
└─controlDownloadPeriodBegin	4	4	{00..00}
└─controlDownloadPeriodEnd	4	4	{00..00}
EF Specific_Conditions	10	10	
└─SpecificConditionRecord	2	5	5
└─entryTime	4	4	{00..00}
└─SpecificConditionType	1	1	{00}

TCS_159 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the workshop card data structure must use for a generation 1 application:

		Min	Max
n1	NoOfEventsPerType	3	3
n2	NoOfFaultsPerType	6	6
n3	NoOfCardVehicleRecords	4	8
n4	NoOfCardPlaceRecords	6	8
n5	NoOfCalibrationRecords	88	255
n6	CardActivityLengthRange	198 bytes (1 day * 93 activity changes)	492 bytes (1 day * 240 activity changes)

4.3.2 Workshop card application generation 2

TCS_160 After its personalisation, the workshop card application generation 2 shall have the following permanent file structure and file access rules.

Notes:

- The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.
- **EF Application_Identification_V2, EF Places_Authentication, EF GNSS_Places_Authentication, EF Border_Crossings, EF Load_Unload_Operations, EF Load_Type_Entries, EF VU_Configuration and EF Calibration_Add_Data** are only present in version 2 of the generation 2 workshop card.
- **cardStructureVersion in EF Application_Identification** is equal to {01 01} for version 2 of the generation 2 workshop card, while it was equal to {01 00} for version 1 of the generation 2 workshop card.

File	File-ID	SFID	Access rules		
			Read	Select	Update
└─DF Tachograph_G2			SC1	SC1	
└─EF Application_Identification	'0501h'	1	SC1	SC1	NEV
└─EF CardMA_Certificate	'C100h'	2	SC1	SC1	NEV
└─EF CardSignCertificate	'C101h'	3	SC1	SC1	NEV
└─EF CA_Certificate	'C108h'	4	SC1	SC1	NEV
└─EF Link_Certificate	'C109h'	5	SC1	SC1	NEV
└─EF Identification	'0520h'	6	SC1	SC1	NEV
└─EF Card_Download	'0509h'	7	SC1	SC1	SC1
└─EF Calibration	'050Ah'	10	SC1	SC1	SM-MAC-G2

EF_Sensor_Installation_Data	'050Bh'	11	SC5	SM-MAC-G2	NEV
EF_Events_Data	'0502h'	12	SC1	SC1	SM-MAC-G2
EF_Faults_Data	'0503h'	13	SC1	SC1	SM-MAC-G2
EF_Driver_Activity_Data	'0504h'	14	SC1	SC1	SM-MAC-G2
EF_Vehicles_Used	'0505h'	15	SC1	SC1	SM-MAC-G2
EF_Places	'0506h'	16	SC1	SC1	SM-MAC-G2
EF_Current_Usage	'0507h'	17	SC1	SC1	SM-MAC-G2
EF_Control_Activity_Data	'0508h'	18	SC1	SC1	SM-MAC-G2
EF_Specific_Conditions	'0522h'	19	SC1	SC1	SM-MAC-G2
EF_VehicleUnits_Used	'0523h'	20	SC1	SC1	SM-MAC-G2
EF_GNSS_Places	'0524h'	21	SC1	SC1	SM-MAC-G2

File	File ID	SFID	Access rules		
			Read	Select	Update
DF_Tachograph_G2			SC1	SC1	
EF_Application_Identification	'0501h'	1	SC1	SC1	NEV
EF_CardMA_Certificate	'C100h'	2	SC1	SC1	NEV
EF_CardSignCertificate	'C101h'	3	SC1	SC1	NEV
EF_CA_Certificate	'C108h'	4	SC1	SC1	NEV
EF_Link_Certificate	'C109h'	5	SC1	SC1	NEV
EF_Identification	'0520h'	6	SC1	SC1	NEV
EF_Card_Download	'0509h'	7	SC1	SC1	SC1
EF_Calibration	'050Ah'	10	SC1	SC1	SM-MAC-G2
EF_Sensor_Installation_Data	'050Bh'	11	SC5	SM-MAC-G2	NEV
EF_Events_Data	'0502h'	12	SC1	SC1	SM-MAC-G2
EF_Faults_Data	'0503h'	13	SC1	SC1	SM-MAC-G2
EF_Driver_Activity_Data	'0504h'	14	SC1	SC1	SM-MAC-G2
EF_Vehicles_Used	'0505h'	15	SC1	SC1	SM-MAC-G2
EF_Places	'0506h'	16	SC1	SC1	SM-MAC-G2
EF_Current_Usage	'0507h'	17	SC1	SC1	SM-MAC-G2
EF_Control_Activity_Data	'0508h'	18	SC1	SC1	SM-MAC-G2
EF_Specific_Conditions	'0522h'	19	SC1	SC1	SM-MAC-G2
EF_VehicleUnits_Used	'0523h'	20	SC1	SC1	SM-MAC-G2
EF_GNSS_Places	'0524h'	21	SC1	SC1	SM-MAC-G2
EF_Application_Identification_V2	'0525h'	22	SC1	SC1	NEV
EF_Places_Authentication	'0526h'	23	SC1	SC1	SM-MAC-G2
EF_GNSS_Places_Authentication	'0527h'	24	SC1	SC1	SM-MAC-G2
EF_Border_Crossings	'0528h'	25	SC1	SC1	SM-MAC-G2
EF_Load_Unload_Operations	'0529h'	26	SC1	SC1	SM-MAC-G2
EF_Load_Type_Entries	'0530h'	27	SC1	SC1	SM-MAC-G2
EF_Calibration_Add_Data	'0531h'	28	SC1	SC1	SM-MAC-G2
EF_VU_Configuration	'0540h'	30	SC5	SC1	SM-MAC-G2

The following abbreviations for the security conditions are used in this table:

SC1 ALW OR SM-MAC-G2

SC5 For the Read Binary command with even INS byte: SM-C-MAC-G2 AND SM-R-ENC-MAC-G2

For the Read Binary command with odd INS byte (if supported): NEV

TCS_161 All EFs structures shall be transparent.

TCS_162 The workshop card application generation 2 shall have the following data structure:

File / Data element	No-of Records	Size (Bytes)		Default Values
		Min	Max	
DF Tachograph_G2	18783	49783	49787	
EF Application_Identification	19	19	19	
L WorkshopCardApplicationIdentification	19	19	19	
typeOfTachographCardId	1	1	1	{00}
cardStructureVersion	2	2	2	{00..00}
noOfEventsPerType	1	1	1	{00}
noOfFaultsPerType	1	1	1	{00}
activityStructureLength	2	2	2	{00..00}
noOfCardVehicleRecords	2	2	2	{00..00}
noOfCardPlaceRecords	2	2	2	{00..00}
noOfCalibrationRecords	2	2	2	{00..00}
noOfGNSSADRecords	2	2	2	{00..00}
noOfSpecificConditionRecords	2	2	2	{00..00}
noOfCardVehicleUnitRecords	2	2	2	{00..00}
EF CardMA_Certificate			341	
L CardMACertificate		204	341	{00..00}
EF CardSignCertificate		204	341	
L CardSignCertificate		204	341	{00..00}
EF CA_Certificate		204	341	
L MemberStateCertificate		204	341	{00..00}
EF Link_Certificate		204	341	
L LinkCertificate		204	341	{00..00}
EF Identification	211	211	211	
L CardIdentification		65	65	
cardIssuingMemberState	1	1	1	{00}
cardNumber	16	16	16	{20..20}
cardIssuingAuthorityName	36	36	36	{00,20..20}
cardIssueDate	4	4	4	{00..00}
cardValidityBegin	4	4	4	{00..00}
cardExpiryDate	4	4	4	{00..00}
L WorkshopCardHolderIdentification		146	146	
workshopName	36	36	36	{00,20..20}
workshopAddress	36	36	36	{00,20..20}
cardHolderName				
holderSurname	36	36	36	{00,20..20}
holderFirstNames	36	36	36	{00,20..20}
cardHolderPreferredLanguage	2	2	2	{20,20}
EF Card_Download		2	2	
L NoOfCalibrationsSinceDownload		2	2	{00..00}
EF Calibration	15668	45394	45394	
L WorkshopCardCalibrationData		15668	45394	
calibrationTotalNumber		2	2	{00..00}
calibrationPointerNewestRecord		2	2	{00}
calibrationRecords		15664	45390	
L WorkshopCardCalibrationRecord	n5	178	178	
calibrationPurpose		1	1	{00}
vehicleIdentificationNumber		17	17	{20..20}
vehicleRegistration				
L				

└─ vehicleRegistrationNation	1	1	{00}
└─ vehicleRegistrationNumber	14	14	{00,20..20}
wVehicleCharacteristicConstant	2	2	{00.00}
kConstantOfRecordingEquipment	2	2	{00.00}
lTyreCircumference	2	2	{00.00}
tyreSize	15	15	{20..20}
authorisedSpeed	1	1	{00}
oldOdometerValue	3	3	{00..00}
newOdometerValue	3	3	{00..00}
oldTimeValue	4	4	{00..00}
newTimeValue	4	4	{00..00}
extCalibrationDate	4	4	{00..00}
vuPartNumber	16	16	{20..20}
vuSerialNumber	8	8	{00..00}
sensorSerialNumber	8	8	{00..00}
sensorGNSSSerialNumber	8	8	{00..00}
remSerialNumber	8	8	{00..00}
vuAbility	1	1	{00}
sealDataCard	56	56	
└─ noOfSealRecords	1	1	{00}
└─ SealRecords	55	55	
└─ SealRecord	5	11	11
└─ equipmentType	1	1	{00}
└─ extendedSealIdentifier	10	10	{00..00}
EF Sensor_Installation_Data	18	102	
└─ SensorInstallationSecData	18	102	{00..00}
EF Events_Data	792	792	
└─ CardEventData	792	792	
└─ CardEventRecords	11	72	72
└─ CardEventRecord	n1	24	24
└─ eventType	1	1	{00}
└─ eventBeginTime	4	4	{00..00}
└─ eventEndTime	4	4	{00..00}
└─ eventVehicleRegistration			
└─ vehicleRegistrationNation	1	1	{00}
└─ vehicleRegistrationNumber	14	14	{00,20..20}
EF Faults_Data	288	288	
└─ CardFaultData	288	288	
└─ CardFaultRecords	2	144	144
└─ CardFaultRecord	n2	24	24
└─ faultType	1	1	{00}
└─ faultBeginTime	4	4	{00..00}
└─ faultEndTime	4	4	{00..00}
└─ faultVehicleRegistration			
└─ vehicleRegistrationNation	1	1	{00}
└─ vehicleRegistrationNumber	14	14	{00,20..20}
EF Driver_Activity_Data	202	496	
└─ CardDriverActivity	202	496	
└─ activityPointerOldestDayRecord	2	2	{00.00}
└─ activityPointerNewestRecord	2	2	{00.00}
└─ activityDailyRecords	n6	198	492
EF Vehicles_Used	194	386	
└─ CardVehiclesUsed	194	386	

└ vehiclePointerNewestRecord		2	2	{00..00}
└ cardVehicleRecords		192	384	
└ CardVehicleRecord	n3	48	48	
└ vehicleOdometerBegin		3	3	{00..00}
└ vehicleOdometerEnd		3	3	{00..00}
└ vehicleFirstUse		4	4	{00..00}
└ vehicleLastUse		4	4	{00..00}
└ vehicleRegistration				
└ vehicleRegistrationNation		1	1	{00}
└ vehicleRegistrationNumber		14	14	{00,20..20}
└ vuDataBlockCounter		2	2	{00..00}
└ vehicleIdentificationNumber		17	17	{20..20}
EF Places		128	170	
└ CardPlaceDailyWorkPeriod		128	170	
└ placePointerNewestRecord		2	2	{00..00}
└ placeRecords		126	168	
└ PlaceRecord	n4	21	21	
└ entryTime		4	4	{00..00}
└ entryTypeDailyWorkPeriod		1	1	{00}
└ dailyWorkPeriodCountry		1	1	{00}
└ dailyWorkPeriodRegion		1	1	{00}
└ vehicleOdometerValue		3	3	{00..00}
└ entryGNSSPlaceRecord		11	11	{00..00}
└ timeStamp		4	4	{00..00}
└ gnssAccuracy		1	1	{00}
└ geoCoordinates		6	6	{00..00}
EF Current_Usage		19	19	
└ CardCurrentUse		19	19	
└ sessionOpenTime		4	4	{00..00}
└ sessionOpenVehicle				
└ vehicleRegistrationNation		1	1	{00}
└ vehicleRegistrationNumber		14	14	{00,20..20}
EF Control_Activity_Data		46	46	
└ CardControlActivityDataRecord		46	46	
└ controlType		1	1	{00}
└ controlTime		4	4	{00..00}
└ controlCardNumber				
└ cardType		1	1	{00}
└ cardIssuingMemberState		1	1	{00}
└ cardNumber		16	16	{20..20}
└ controlVehicleRegistration				
└ vehicleRegistrationNation		1	1	{00}
└ vehicleRegistrationNumber		14	14	{00,20..20}
└ controlDownloadPeriodBegin		4	4	{00..00}
└ controlDownloadPeriodEnd		4	4	{00..00}
EF VehicleUnits_Used		42	42	
└ CardVehicleUnitsUsed		42	82	
└ vehicleUnitPointerNewestRecord		2	2	{00..00}
└ cardVehicleUnitRecords		40	80	
└ CardVehicleUnitRecord	n7	10	10	
└ timeStamp		4	4	{00..00}
└ manufacturerCode		1	1	{00..00}

deviceID		1	1	{00..00}
vuSoftwareVersion		4	4	{00..00}
EF GNSS_Places		326	432	
GNSSContinuousDriving		326	434	
gnssADPointerNewestRecord		2	2	{00..00}
gnssAccumulatedDrivingRecords		324	432	
GNSSContinuousDrivingRecord	n8	18	18	
timeStamp		4	4	{00..00}
gnssPlaceRecord		14	14	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
vehicleOdometerValue		3	3	{00..00}
EF Specific_Conditions		12	22	
SpecificConditions		12	22	
conditionPointerNewestRecord		2	2	{00..00}
specificConditionRecords		10	20	
SpecificConditionRecord	n9	5	5	
entryTime		4	4	{00..00}
specificConditionType		1	1	{00}

File / Data Element	No of Records	Size (bytes) Min	Size (bytes) Max	Default Values
DF		59582	60214	
Tachograph_G2		59582	60214	
EF Application_Identification		19	19	
WorkshopCardApplicationIdentification		19	19	
typeOfTachographCardId		1	1	{00}
cardStructureVersion		2	2	{01 01}
noOfEventsPerType		1	1	{00}
noOfFaultsPerType		1	1	{00}
activityStructureLength		2	2	{00 00}
noOfCardVehicleRecords		2	2	{00 00}
noOfCardPlaceRecords		2	2	{00 00}
noOfCalibrationRecords		2	2	{00 00}
noOfGNSSADRecords		2	2	{00 00}
noOfSpecificConditionRecords		2	2	{00 00}
noOfCardVehicleUnitRecords		2	2	{00 00}
EF CardMA_Certificate		204	341	
CardMA_Certificate		204	341	{00..00}
EF CardSignCertificate		204	341	
CardSignCertificate		204	341	{00..00}
EF CA_Certificate		204	341	
MemberStateCertificate		204	341	{00..00}
EF Link_Certificate		204	341	
LinkCertificate		204	341	{00..00}
EF Identification		211	211	
CardIdentification		65	65	
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}
cardIssuingAuthorityName		36	36	{00, 20..20}
cardIssueDate		4	4	{00..00}
cardValidityBegin		4	4	{00..00}

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
cardExpiryDate		4	4	{00..00}
WorkshopCardHolderIdentification		146	146	
workshopName		36	36	
workshopAddress		36	36	
cardHolderName		72	72	
holderSurname		36	36	{00, 20..20}
holderFirstNames		36	36	{00, 20..20}
cardHolderPreferredLanguage		2	2	{20 20}
EF Card_Download		2	2	
NoOfCalibrationsSinceDownload		2	2	{00 00}
EF Calibration		45394	45394	
WorkshopCardCalibrationData		45394	45394	
calibrationTotalNumber		2	2	{00 00}
calibrationPointerNewestRecord		2	2	{00 00}
calibrationRecords		45390	45390	
WorkshopCardCalibrationRecord	n5	178	178	
calibrationPurpose		1	1	{00}
vehicleIdentificationNumber		17	17	{20..20}
vehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
wVehicleCharacteristicConstant		2	2	{00 00}
kConstantOfRecordingEquipment		2	2	{00 00}
lTyreCircumference		2	2	{00 00}
tyreSize		15	15	{20..20}
authorisedSpeed		1	1	{00}
oldOdometerValue		3	3	{00..00}
newOdometerValue		3	3	{00..00}
oldTimeValue		4	4	{00..00}
newTimeValue		4	4	{00..00}
nextCalibrationDate		4	4	{00..00}
vuPartNumber		16	16	{20..20}
vuSerialNumber		8	8	{00..00}
sensorSerialNumber		8	8	{00..00}
sensorGNSSSerialNumber		8	8	{00..00}
rcmSerialNumber		8	8	{00..00}
vuAbility		1	1	{00}
sealDataCard		56	56	
noOfSealRecords		1	1	{00}
SealRecords		55	55	
SealRecord	5	11	11	
equipmentType		1	1	{00}
extendedSealIdentifier		10	10	{00..00}
EF Sensor_Installation_Data		18	102	
SensorInstallationSecData		18	102	{00..00}
EF Events_Data		792	792	
CardEventData		792	792	
cardEventRecords	11	72	72	
CardEventRecord	n1	24	24	
eventType		1	1	{00}
eventBeginTime		4	4	{00..00}
eventEndTime		4	4	{00..00}
eventVehicleRegistration				

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
				vehicleRegistrationNation
				vehicleRegistrationNumber
EF Faults_Data		288	288	
CardFaultData		288	288	
cardFaultRecords	2	144	144	
CardFaultRecord	n2	24	24	
faultType		1	1	{00}
faultBeginTime		4	4	{00..00}
faultEndTime		4	4	{00..00}
faultVehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
EF Driver Activity_Data		496	496	
CardDriverActivity		496	496	
activityPointerOldestDayRecord		2	2	{00 00}
activityPointerNewestRecord		2	2	{00 00}
activityDailyRecords	n6	492	492	{00..00}
EF Vehicles_Used		386	386	
CardVehiclesUsed		386	386	
vehiclePointerNewestRecord		2	2	{00 00}
cardVehicleRecords		384	384	
cardVehicleRecord	n3	48	48	
vehicleOdometerBegin		3	3	{00..00}
vehicleOdometerEnd		3	3	{00..00}
vehicleFirstUse		4	4	{00..00}
vehicleLastUse		4	4	{00..00}
vehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
vuDataBlockCounter		2	2	{00 00}
vehicleIdentificationNumber		17	17	{20..20}
EF Places		170	170	
CardPlaceDailyWorkPeriod		170	170	
placePointerNewestRecord		2	2	{00 00}
placeRecords		168	168	
PlaceRecord	n4	21	21	
entryTime		4	4	{00..00}
entryTypeDailyWorkPeriod		1	1	{00}
dailyWorkPeriodCountry		1	1	{00}
dailyWorkPeriodRegion		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
entryGNSSPlaceRecord		11	11	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
EF Current_Usage		19	19	
CardCurrentUse		19	19	
sessionOpenTime		4	4	{00..00}
sessionOpenVehicle				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
EF Control_Activity_Data		46	46	
CardControlActivityDataRecord		46	46	

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
controlType		1	1	{00}
controlTime		4	4	{00..00}
controlCardNumber				
cardType		1	1	{00}
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}
controlVehicleRegistration				
vehicleRegistrationNation		1	1	{00}
vehicleRegistrationNumber		14	14	{00, 20..20}
controlDownloadPeriodBegin		4	4	{00..00}
controlDownloadPeriodEnd		4	4	{00..00}
EF VehicleUnits_Used		82	82	
CardVehicleUnitsUsed		82	82	
vehicleUnitPointerNewestRecord		2	2	{00 00}
cardVehicleUnitRecords		80	80	
CardVehicleUnitRecord	n7	10	10	
timeStamp		4	4	{00..00}
manufacturerCode		1	1	{00}
deviceID		1	1	{00}
vuSoftwareVersion		4	4	{00..00}
EF GNSS_Places		434	434	
GNSSAccumulatedDriving		434	434	
gnssADPointerNewestRecord		2	2	{00 00}
gnssAccumulatedDrivingRecords		432	432	
GNSSAccumulatedDrivingRecord	n8	18	18	
timeStamp		4	4	{00..00}
gnssPlaceRecord		14	14	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
vehicleOdometerValue		3	3	{00..00}
EF Specific_Conditions		22	22	
SpecificConditions		22	22	
conditionPointerNewestRecord		2	2	{00 00}
specificConditionRecords		20	20	
SpecificConditionRecord	n9	5	5	
entryTime		4	4	{00..00}
specificConditionType		1	1	{00}
EF Application_Identification_V2		10	10	
WorkshopCardApplicationIdentificationV2		10	10	
LengthOfFollowingData		2	2	{00 00}
noOfBorderCrossingRecords		2	2	{00 00}
noOfLoadUnloadRecords		2	2	{00 00}
noOfLoadTypeEntryRecords		2	2	{00 00}
VuConfigurationLengthRange		2	2	{00 00}
EF Places_Authentication		42	42	
CardPlaceAuthDailyWorkPeriod		42	42	
placeAuthPointerNewestRecord		2	2	{00 00}
placeAuthStatusRecords		40	40	
PlaceAuthStatusRecord	n4	5	5	
entryTime		4	4	{00..00}
authenticationStatus		1	1	{00}
EF GNSS_Places_Authentication		122	122	

File / Data Element	No of Records	Size (bytes)		Default Values
		Min	Max	
GNSSAuthAccumulatedDriving		122	122	
gnssAuthADPointerNewestRecord		2	2	{00 00}
gnssAuthStatusADRecords		120	120	
GNSSAuthStatusADRecord	n8	5	5	
timeStamp		4	4	{00..00}
authenticationStatus		1	1	{00}
EF Border_Crossings		70	70	
CardBorderCrossings		70	70	
borderCrossingPointerNewestRecord		2	2	{00 00}
cardBorderCrossingRecords		68	68	
CardBorderCrossingRecord	n10	17	17	
countryLeft		1	1	{00}
countryEntered		1	1	{00}
gnssPlaceAuthRecord		12	12	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
authenticationStatus		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
EF Load_Unload_Operations		162	162	
CardLoadUnloadOperations		162	162	
loadUnloadPointerNewestRecord		2	2	{00 00}
cardloadUnloadRecords		160	160	
CardLoadUnloadRecord	n11	20	20	
timestamp		4	4	{00}
operationType		1	1	{00..00}
gnssPlaceAuthRecord		12	12	
timeStamp		4	4	{00..00}
gnssAccuracy		1	1	{00}
geoCoordinates		6	6	{00..00}
authenticationStatus		1	1	{00}
vehicleOdometerValue		3	3	{00..00}
EF Load_Type_Entries		22	22	
CardLoadTypeEntries		22	22	
loadtypeEntryPointerNewestRecord		2	2	{00 00}
cardLoadTypeEntryRecords		20	20	
CardLoadTypeEntryRecord	n12	5	5	
timestamp		4	4	{00..00}
loadTypeEntered		1	1	{00}
EF Calibration_Add_Data		6887	6887	
WorkshopCardCalibrationAddData		6887	6887	
calibrationPointerNewestRecord		2	2	{00 00}
workshopCardCalibrationAddDataRecords		6885	6885	
WorkshopCardCalibrationAddDataRecord	n5	27	27	
oldTimeValue		4	4	{00..00}
vehicleIdentificationNumber		17	17	{20..20}
byDefaultLoadType		1	1	{00}
calibrationCountry		1	1	{00}
calibrationCountryTimestamp		4	4	{00..00}
EF VU_Configuration		3072	3072	
VuConfigurations	n13	3072	3072	

TCS_163 The following values, used to provide sizes in the table above, are the minimum and maximum record –number values the workshop card data structure must use for a generation 2 application:

		<i>Min</i>	<i>Max</i>
n ₁	NoOfEventsPerType	3	3
n ₂	NoOfFaultsPerType	6	6
n ₃	NoOfCardVehicleRecords	8 48	8
n ₄	NoOfCardPlaceRecords	8 68	8
n ₅	NoOfCalibrationRecords	255 88255	255
n ₆	CardActivityLengthRange	492 492 bytes (1 day * 240 240 activity changes)	492 bytes (1 day * 240 activity changes)
n ₇	NoOfCardVehicleUnitRecords	8 48	8
n ₈	NoOfGNSSCDRecords NoOfGNSSADRecords	24 1824	24
n ₉	NoOfSpecificConditionRecords	4 24	4
n ₁₀	NoOfBorderCrossingRecords	4 4	4 4
n ₁₁	NoOfLoadUnloadRecords	8 8	8 8
n ₁₂	NoOfLoadTypeEntryRecords	4 4	4 4
n ₁₃	VuConfigurationLengthRange	3072 3072 Bytes	3072 3072 Bytes

4.4. Control card applications

4.4.1 Control Card application generation 1

TCS_164 After its personalisation, the control card application generation 1 shall have the following permanent file structure and file access rules:

File	File ID	Access rules		
		Read	Select	Update
└DF Tachograph	'0500h'			
└└EF Application_Identification	'0501h'	SC2	SC1	NEV
└└EF Card_Certificate	'C100h'	SC2	SC1	NEV
└└EF CA_Certificate	'C108h'	SC2	SC1	NEV
└└EF Identification	'0520h'	SC6	SC1	NEV
└└EF Controller_Activity_Data	'050Ch'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

- SC1** ALW OR SM-MAC-G2
SC2 ALW OR SM-MAC-G1 OR SM-MAC-G2
SC3 SM-MAC-G1 OR SM-MAC-G2
SC6 EXT-AUT-G1 OR SM-MAC-G1 OR SM-MAC-G2

TCS_165 All EF structures shall be transparent.

TCS_166 The control card application generation 1 shall have the following data structure:

File / Data element	No of Records	Size (Bytes)	
		Min	Max
└ DF Tachograph		11186	24526
└ EF Application_Identification		5	5
└└ ControlCardApplicationIdentification		5	5
└└└ typeOfTachographCardId		1	1 {00}
└└└ cardStructureVersion		2	2 {00 00}
└└└ noOfControlActivityRecords		2	2 {00 00}
└ EF Card_Certificate		194	194
└└ CardCertificate		194	194 {00..00}
└ EF CA_Certificate		194	194
└└ MemberStateCertificate		194	194 {00..00}
└ EF Identification		211	211
└└ CardIdentification		65	65
└└└ cardIssuingMemberState		1	1 {00}
└└└ cardNumber		16	16 {20..20}
└└└ cardIssuingAuthorityName		36	36 {00, 20..20}
└└└ cardIssueDate		4	4 {00..00}
└└└ cardValidityBegin		4	4 {00..00}
└└└ cardExpiryDate		4	4 {00..00}
└└ ControlCardHolderIdentification		146	146
└└└ controlBodyName		36	36 {00, 20..20}
└└└ controlBodyAddress		36	36 {00, 20..20}
└└└ cardHolderName			
└└└└ holderSurname		36	36 {00, 20..20}
└└└└ holderFirstNames		36	36 {00, 20..20}
└└└ cardHolderPreferredLanguage		2	2 {20 20}
└ EF Controller_Activity_Data		10582	23922
└└ ControlCardControlActivityData		10582	23922
└└└ controlPointerNewestRecord		2	2 {00 00}
└└└ controlActivityRecords		10580	23920
└└└└ controlActivityRecord	n7	46	46
└└└└└ controlType		1	1 {00}
└└└└└ controlTime		4	4 {00..00}
└└└└└ controlledCardNumber			
└└└└└└ cardType		1	1 {00}
└└└└└└ cardIssuingMemberState		1	1 {00}
└└└└└└ cardNumber		16	16 {20..20}
└└└└└ controlledVehicleRegistration			
└└└└└└ vehicleRegistrationNation		1	1 {00}
└└└└└└ vehicleRegistrationNumber		14	14 {00, 20..20}
└└└└└ controlDownloadPeriodBegin		4	4 {00..00}
└└└└└ controlDownloadPeriodEnd		4	4 {00..00}

TCS_167 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the control card data structure must use for a generation 1 application:

		<i>Min</i>	<i>Max</i>
n7	NoOfControlActivityRecords	230	520

4.4.2 Control card application generation 2

TCS_168 After its personalisation, the control card application generation 2 shall have the following permanent file structure and file access rules.

Notes: -

- The short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.
- **EF Application_Identification_V2, and EF VU_Configuration are only present in version 2 of the generation 2 control card,**
- **cardStructureVersion in EF Application_Identification is equal to {01 01} for version 2 of the generation 2 control card, while it was equal to {01 00} for version 1 of the generation 2 control card.**

File	File-ID	SFID	Access rules	
			Read / Select	Update
└DF Tachograph_G2			SC1	
└EF Application_Identification	'0501h'	1	SC1	NEV
└EF CardMA_Certificate	'C100h'	2	SC1	NEV
└EF CA_Certificate	'C108h'	4	SC1	NEV
└EF Link_Certificate	'C109h'	5	SC1	NEV
└EF Identification	'0520h'	6	SC1	NEV
└EF Controller_Activity_Data	'050Ch'	14	SC1	SM-MAC-G2

File	File ID	SFID	Access rules	
			Read / Select	Update
└DF Tachograph_G2			SC1	
└EF Application_Identification	'0501h'	1	SC1	NEV
└EF CardMA_Certificate	'C100h'	2	SC1	NEV
└EF CA_Certificate	'C108h'	4	SC1	NEV
└EF Link_Certificate	'C109h'	5	SC1	NEV
└EF Identification	'0520h'	6	SC1	NEV
└EF Controller_Activity_Data	'050Ch'	14	SC1	SM-MAC-G2
└EF Application_Identification_V2	'0525h'	22	SC1	NEV
└EF VU_Configuration	'0540h'	30	SC5/SC1	SM-MAC-G2

The following abbreviations for the security condition are used in this table:

SC1 ALW OR SM-MAC-G2

SC5 For the Read Binary command with even INS byte: SM-C-MAC-G2 AND SM-R-ENC-MAC-G2
 For the Read Binary command with odd INS byte (if supported): NEV

TCS_169 All EF structures shall be transparent.

TCS_170 The control card application generation2 shall have the following data structure:

File / Data element	No of Records	Size (Bytes)	
		Min	Max
└DF Tachograph_G2		11410	25161
└EF Application_Identification		5	5
└ControlCardApplicationIdentification		5	5
└typeOfTachographCardId	1	1	{00}

└ cardStructureVersion	2	2	{00..00}
└ noOfControlActivityRecords	2	2	{00..00}
EF CardMA_Certificate	204	341	
└ CardMACertificate	204	341	{00..00}
EF CA_Certificate	204	341	
└ MemberStateCertificate	204	341	{00..00}
EF Link_Certificate	204	341	
└ LinkCertificate	204	341	{00..00}
EF Identification	211	211	
└ CardIdentification	65	65	
└ cardIssuingMemberState	1	1	{00}
└ cardNumber	16	16	{20..20}
└ cardIssuingAuthorityName	36	36	{00,20..20}
└ cardIssueDate	4	4	{00..00}
└ cardValidityBegin	4	4	{00..00}
└ cardExpiryDate	4	4	{00..00}
└ ControlCardHolderIdentification	146	146	
└ controlBodyName	36	36	{00,20..20}
└ controlBodyAddress	36	36	{00,20..20}
└ cardHolderName			
└ holderSurname	36	36	{00,20..20}
└ holderFirstNames	36	36	{00,20..20}
└ cardHolderPreferredLanguage	2	2	{20 20}
EF Controller_Activity_Data	10582	23922	
└ ControlCardControlActivityData	10582	23922	
└ controlPointerNewestRecord	2	2	{00..00}
└ controlActivityRecords	10580	23920	
└ controlActivityRecord	n7	46	46
└ controlType	1	1	{00}
└ controlTime	4	4	{00..00}
└ controlledCardNumber			
└ cardType	1	1	{00}
└ cardIssuingMemberState	1	1	{00}
└ cardNumber	16	16	{20..20}
└ controlledVehicleRegistration			
└ vehicleRegistrationNation	1	1	{00}
└ vehicleRegistrationNumber	14	14	{00,20..20}
└ controlDownloadPeriodBegin	4	4	{00..00}
└ controlDownloadPeriodEnd	4	4	{00..00}

File / Data Element	No of Records	Min	Max	Default Values
DF Tachograph_G2		14486	28237	
EF Application_Identification		5	5	
ControlCardApplicationIdentification		5	5	
typeOfTachographCardId		1	1	{00}
cardStructureVersion		2	2	{01 01} V2
noOfControlActivityRecords		2	2	{00 00}
EF CardMA_Certificate		204	341	

CardMA_Certificate	204	341	{00..00}
EF CA_Certificate	204	341	
MemberStateCertificate	204	341	{00..00}
EF Link_Certificate	204	341	
LinkCertificate	204	341	{00..00}
EF Identification	211	211	
CardIdentification	65	65	
cardIssuingMemberState	1	1	{00}
cardNumber	16	16	{20..20}
cardIssuingAuthorityName	36	36	{00, 20..20}
cardIssueDate	4	4	{00..00}
cardValidityBegin	4	4	{00..00}
cardExpiryDate	4	4	{00..00}
ControlCardHolderIdentification	146	146	
controlBodyName	36	36	{00, 20..20}
controlBodyAddress	36	36	{00, 20..20}
cardHolderName			
holderSurname	36	36	{00, 20..20}
holderFirstNames	36	36	{00, 20..20}
cardHolderPreferredLanguage	2	2	{20 20}
EF Controller_Activity_Data	10582	23922	
ControlCardControlActivityData	10582	23922	
controlPointerNewestRecord	2	2	{00 00}
controlActivityRecords	10580	23920	
controlActivityRecord	n7	46	46
controlType	1	1	{00}
controlTime	4	4	{00..00}
controlledCardNumber			
cardType	1	1	{00}
cardIssuingMemberState	1	1	{00}
cardNumber	16	16	{20..20}
controlledVehicleRegistration			
vehicleRegistrationNation	1	1	{00}
vehicleRegistrationNumber	14	14	{00, 20..20}
controlDownloadPeriodBegin	4	4	{00..00}
controlDownloadPeriodEnd	4	4	{00..00}
EF Application_Identification_V2	4	4	
ControlCardApplicationIdentificationV2	4	4	
lengthOfFollowingData	2	2	{00 00}
VuConfigurationLengthRange	2	2	{00 00}
EF VuConfiguration	3072	3072	
VuConfigurations	n13	3072	3072

TCS_171 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the control card data structure must use for a generation 2 application:

		<i>Min</i>	<i>Max</i>
n7	NoOfControlActivityRecords	230	520
n13	VuConfigurationLengthRange	3072 Bytes	3072 Bytes

4.5. Company card applications

4.5.1 Company card application generation 1

TCS_172 After its personalisation, the company card application generation 1 shall have the following permanent file structure and file access rules:

File	File ID	Access rules		
		Read	Select	Update
└DF Tachograph	'0500h'		SC1	
└EF Application Identification	'0501h'	SC2	SC1	NEV
└EF Card Certificate	'C100h'	SC2	SC1	NEV
└EF CA_Certificate	'C108h'	SC2	SC1	NEV
└EF Identification	'0520h'	SC6	SC1	NEV
└EF Company_Activity_Data	'050Dh'	SC2	SC1	SC3

The following abbreviations for the security conditions are used in this table:

SC1 ALW OR SM-MAC-G2

SC2 ALW OR SM-MAC-G1 OR SM-MAC-G2

SC3 SM-MAC-G1 OR SM-MAC-G2

SC6 EXT-AUT-G1 OR SM-MAC-G1 OR SM-MAC-G2

TCS_173 All EF structures shall be transparent.

TCS_174 The company card application generation 1 shall have the following data structure:

File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
└DF Tachograph		11114	24454	
└EF Application Identification		5	5	
└└CompanyCardApplicationIdentification		5	5	
└└└typeOfTachographCardId		1	1	{00}
└└└cardStructureVersion		2	2	{00.00}
└└└noOfCompanyActivityRecords		2	2	{00.00}
└EF Card_Certificate		194	194	
└└CardCertificate		194	194	{00..00}
└EF CA_Certificate		194	194	
└└MemberStateCertificate		194	194	{00..00}
└EF Identification		139	139	
└└CardIdentification		65	65	
└└└cardIssuingMemberState		1	1	{00}
└└└cardNumber		16	16	{20..20}
└└└cardIssuingAuthorityName		36	36	{00, 20..20}
└└└cardIssueDate		4	4	{00..00}
└└└cardValidityBegin		4	4	{00..00}
└└└cardExpiryDate		4	4	{00..00}
└└CompanyCardHolderIdentification		74	74	
└└└companyName		36	36	{00, 20..20}
└└└companyAddress		36	36	{00, 20..20}

EF cardHolderPreferredLanguage	2	2	{00..00}
EF Company_Activity_Data	10582	23922	
└ CompanyActivityData	10582	23922	
└└ companyPointerNewestRecord	2	2	{00..00}
└└ companyActivityRecords	10580	23920	
└└└ companyActivityRecord	n ₈	46	46
└└└└ companyActivityType	1	1	{00}
└└└└ companyActivityTime	4	4	{00..00}
└└└└ cardNumberInformation			
└└└└└ cardType	1	1	{00}
└└└└└ cardIssuingMemberState	1	1	{00}
└└└└└ cardNumber	16	16	{20..20}
└└└└ vehicleRegistrationInformation			
└└└└└ vehicleRegistrationNation	1	1	{00}
└└└└└ vehicleRegistrationNumber	14	14	{00, 20..20}
└└└└ downloadPeriodBegin	4	4	{00..00}
└└└└ downloadPeriodEnd	4	4	{00..00}

TCS_175 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the company card data structure must use for a generation 1 application:

	Min	Max
n ₈ NoOfCompanyActivityRecords	230	520

4.5.2 Company card application generation 2

TCS_176 After its personalisation, the company card application generation 2 shall have the following permanent file structure and file access rules.

Notes: -

- the short EF identifier SFID is given as decimal number, e.g. the value 30 corresponds to 11110 in binary.
- EF Application_Identification_V2, and EF VU_Configuration are only present in version 2 of the generation 2 company card,
- cardStructureVersion in EF Application_Identification is equal to {01 01} for version 2 of the generation 2 company card, while it was equal to {01 00} for version 1 of the generation 2 company card.

File	File ID	SFID	Access rules	
			Read / Select	Update
└ DF Tachograph_G2			SC1	
└└ EF Application_Identification	'0501h'	1	SC1	NEV
└└ EF CardMA_Certificate	'C100h'	2	SC1	NEV
└└ EF CA_Certificate	'C108h'	4	SC1	NEV
└└ EF Link_Certificate	'C109h'	5	SC1	NEV
└└ EF Identification	'0520h'	6	SC1	NEV
└└ EF Company_Activity_Data	'050Dh'	14	SC1	SM MAC G2

File	File ID	SFID	Access rules	
			Read / Select	Update
└ DF Tachograph_G2			SC1	
└└ EF Application_Identification	'0501h'	1	SC1	NEV
└└ EF CardMA_Certificate	'C100h'	2	SC1	NEV

EF CA_Certificate	'C108h'	4	SC1	NEV
EF Link_Certificate	'C109h'	5	SC1	NEV
EF Identification	'0520h'	6	SC1	NEV
EF Company_Activity_Data	'050Dh'	14	SC1	SM-MAC-G2
EF Application_Identification_V2	'0525h'	22	SC1	NEV
EF VU_Configuration	'0540h'	30	SC5/SC1	SM-MAC-G2

The following abbreviations for the security condition are used in this table:

SC1 ALW OR SM-MAC-G2

SC5 For the Read Binary command with even INS byte: SM-C-MAC-G2 AND SM-R-ENC-MAC-G2

For the Read Binary command with odd INS byte (if supported): NEV

TCS_177 All EF structures shall be transparent.

TCS_178 The company card application generation 2 shall have the following data structure:

File / Data element	No of Records	Size (bytes)		Default Values
		Min	Max	
DF Tachograph_G2		11338	25089	
EF Application_Identification		5	5	
L CompanyCardApplicationIdentification		5	5	
typeOfTachographCardId		1	1	{00}
cardStructureVersion		2	2	{00..00}
noOfCompanyActivityRecords		2	2	{00..00}
EF CardMA_Certificate		204	341	
L CardMACertificate		204	341	{00..00}
EF CA_Certificate		204	341	
L MemberStateCertificate		204	341	{00..00}
EF Link_Certificate		204	341	
L LinkCertificate		204	341	{00..00}
EF Identification		139	139	
L CardIdentification		65	65	
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}
cardIssuingAuthorityName		36	36	{00, 20..20}
cardIssueDate		4	4	{00..00}
cardValidityBegin		4	4	{00..00}
cardExpiryDate		4	4	{00..00}
L CompanyCardHolderIdentification		74	74	
companyName		36	36	{00, 20..20}
companyAddress		36	36	{00, 20..20}
cardHolderPreferredLanguage		2	2	{20..20}
EF Company_Activity_Data		10582	23922	
L CompanyActivityData		10582	23922	
companyPointerNewestRecord		2	2	{00..00}
L companyActivityRecords		10580	23920	
L companyActivityRecord	n8	46	46	
companyActivityType		1	1	{00}
companyActivityTime		4	4	{00..00}
L cardNumberInformation				
cardType		1	1	{00}
cardIssuingMemberState		1	1	{00}
cardNumber		16	16	{20..20}

vehicleRegistrationInformation	1	1	{00}
vehicleRegistrationNation	14	14	{00,20..20}
vehicleRegistrationNumber	4	4	{00..00}
downloadPeriodBegin	4	4	{00..00}
downloadPeriodEnd	4	4	{00..00}

File / Data Element	No of Records	Min	Max	Default Values
DF Tachograph_G2	14414	28165		
EF Application_Identification	5	5		
CompanyCardApplicationIdentification	5	5		
typeOfTachographCardId	1	1	{00}	
cardStructureVersion	2	2	{01 01} V2	
noOfCompanyActivityRecords	2	2	{00 00}	
EF CardMA_Certificate	204	341		
CardMA_Certificate	204	341	{00..00}	
EF CA_Certificate	204	341		
MemberStateCertificate	204	341	{00..00}	
EF Link_Certificate	204	341		
LinkCertificate	204	341	{00..00}	
EF Identification	139	139		
CardIdentification	65	65		
cardIssuingMemberState	1	1	{00}	
cardNumber	16	16	{20..20}	
cardIssuingAuthorityName	36	36	{00, 20..20}	
cardIssueDate	4	4	{00..00}	
cardValidityBegin	4	4	{00..00}	
cardExpiryDate	4	4	{00..00}	
CompanyCardHolderIdentification	74	74		
companyName	36	36	{00, 20..20}	
companyAddress	36	36	{00, 20..20}	
cardHolderPreferredLanguage	2	2	{20 20}	
EF Company_Activity_Data	10582	23922		
CompanyActivityData	10582	23922		
companyPointerNewestRecord	2	2	{00 00}	
companyActivityRecords	10580	23920		
companyActivityRecord	n8	46	46	
companyActivityType	1	1	{00}	
companyActivityTime	4	4	{00..00}	
cardNumberInformation				
cardType	1	1	{00}	
cardIssuingMemberState	1	1	{00}	
cardNumber	16	16	{20..20}	
vehicleRegistrationInformation				

	vehicleRegistrationNation	1	1	{00}
	vehicleRegistrationNumber	14	14	{00, 20..20}
	downloadPeriodBegin	4	4	{00..00}
	downloadPeriodEnd	4	4	{00..00}
EF	Application_Identification_V2	4	4	
	CompanyCardApplicationIdentificationV2	4	4	
	lengthOfFollowingData	2	2	{00 00}
	VuConfigurationLengthRange	2	2	{00 00}
EF	VuConfiguration	3072	3072	
	VuConfigurations	n13	3072	3072



































TCS_179 The following values, used to provide sizes in the table above, are the minimum and maximum record number values the company card data structure must use for a generation 2 application:

		<i>Min</i>	<i>Max</i>
n8	NoOfCompanyActivityRecords	230	520
n13	VuConfigurationLengthRange	3072 Bytes	3072 Bytes

Sub-appendix 3. Pictograms

PIC_001 The tachograph may optionally use the following pictograms and pictogram combinations (or pictograms and combination similar enough to be unambiguously identifiable with these):

1. Basic pictograms

	<u>People</u>	<u>Actions</u>	<u>Modes of operation</u>
	Company		Company mode
	Controller	Control	Control mode
	Driver	Driving	Operational mode
	Workshop/test station	Inspection/calibration	Calibration mode
	Manufacturer		
	<u>Activities</u>	<u>Duration</u>	
	Available	Current availability period	
	Driving	Continuous driving time	
	Rest	Current rest period	
	Other Work	Current work period	
	Break	Cumulative break time	
	Unknown		
	<u>Equipment</u>	<u>Functions</u>	
	Driver slot		
	Co-driver slot		
	Card		
	Clock		
	Display	Displaying	
	External storage	Downloading	
	Power supply		
	Printer/printout	Printing	
	Sensor		
	Tyre size		
	Vehicle/vehicle unit		
	GNSS facility		
	Remote Detection facility		
	ITS interface		
	<u>Specific conditions, manual entries</u>		
	Out of scope		
	Ferry/train crossing		
	Load operation		
	Unload operation		
	Simultaneous load/unload operation		
	Load type: passengers		
	Load type: goods		
	Load type: undefined load type		
	<u>Miscellaneous</u>		
	Events	×	Faults

	Start of daily work period		End of daily work period
	Location		
	Manual entry of driver activities		
	Security/authenticated data/seals		
	Speed		
	Time		
	Total/summary		
	Digital map/border crossing		

Qualifiers

24h	Daily
	Weekly
	Two weeks
+	From or to

2. Pictogram combinations

Miscellaneous

	Control place		
	Location start of daily work period		Location end of daily work period
	Position after 3 hours accumulated driving time		
	From time		To time
	From vehicle		
OUT+	Out of scope begin	+OUT	Out of scope end
	Position where the vehicle has crossed the border between two countries		
	Position where a load operation has occurred		
	Position where an unload operation has occurred		
	Position where a simultaneous load/unload operation has occurred		

Cards

	Driver card
	Company card
	Control card
	Workshop card
	No card

Driving

	Crew driving
	Driving time for one week
	Driving time for two weeks

Printouts

24h	Driver activities from card daily printout
24h	Driver activities from VU daily printout
	Events and faults from card printout
	Events and faults from VU printout
	Technical data printout
	Over speeding printout

Historic of inserted cards printout**Events**

- ! Insertion of a non valid card
- ! Card conflict
- ! Time overlap
- ! Driving without an appropriate card
- ! Card insertion while driving
- ! Last card session not correctly closed
- >> Over speeding
- ! Power supply interruption
- ! Motion data error
- ! Vehicle motion conflict
- ! Security breach
- ! Time **conflict or time** adjustment (by workshop)
- > Over speeding control
- ! **Absence of position information from GNSS receiver or Communication error with the external GNSS facility**
- ! **Communication error with the remote communication facility**
- ! **GNSS anomaly**

Faults

- × Card fault (driver slot)
- × Card fault (co-driver slot)
- × Display fault
- × Downloading fault
- × Printer fault
- × Sensor fault
- × VU internal fault

× GNSS fault

× Remote Detection fault

Manual entries procedure

- Still same daily work period ?
- End of previous work period ?
- Confirm or enter location of end of work period
- Enter start time
- Enter location of start of work period.

Note: Additional pictogram combinations to form printout blocks or record identifiers are defined in ~~Appendix~~ **Sub-appendix 4**.

Sub-Appendix 4. Printouts

TABLE OF CONTENT

1. Generalities	312
2. Data blocks specification.....	313
3. Printout specifications.....	321
3.1. Driver Activities from Card Daily Printout.....	322
3.2. Driver Activities from VU Daily Printout.....	323
3.3. Events and Faults from Card Printout.....	324
3.4. Events and Faults from VU Printout.....	324
3.5. Technical data Printout.....	324
3.6. Over speeding Printout	324
3.7. History of Inserted Cards.....	325

1. Generalities

Each printout is built up by chaining various data blocks, possibly identified with a block identifier.

A data block contains one or more records, possibly identified with a record identifier.


- PRT_001 When a block identifier immediately precedes a record identifier, the record identifier is not printed.
- PRT_002 In the case where a data item is unknown, or must not be printed for data access rights reasons, spaces are printed instead.
- PRT_003 If the content of a complete line is unknown, or need not to be printed, then the complete line is omitted.
- PRT_004 Numerical data fields are printed right aligned, with a space separator for thousands and millions, and without leading zeros.
- PRT_005 String data fields are printed left aligned and filled up with spaces to data item length, or truncated to data item length when needed. (~~Names~~ **Names and addresses may be printed in two lines**).
- PRT_006 In case of a line-break due to a long text a special character (dot at middle line-height, ".") should be printed as first character in the new line.

2. Data blocks specification

- In this chapter the following format notation conventions have been used:
- Characters printed in bold denote plain text to be printed (printing remains in normal characters),
- Normal characters denote variables (pictograms or data) to be replaced by their values for printing,
- Variable names have been padded with underscores to show the data item length available for the variable,
- Dates are specified with a “dd/mm/yyyy” (day, month, year) format. A “dd.mm.yyyy” format may also be used,
- The term “card identification” denotes the composition of: the type of card through a card pictograms combination, the card issuing ~~Member State~~ **Contracting Party** code, a forward slash character and the card number with the replacement index and the renewal index separated with a space

P		x	x	x	/	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x			
Card Pictogram combination	Issuing State Party code	Member Contracting Party code	First 14 characters of card number (possibly including a consecutive index)																		Replacement index	Renewal index				

- In a data block, the text after ‘pi=’ refers to the corresponding pictogram or pictogram combination defined in Sub-appendix 3,

- **When printed after the longitude and the latitude of a recorded position, or after the timestamp when the position was determined, the  pictogram indicates that this position has been computed from authenticated navigation messages,**
- *** data only available in GEN2 tachographs (all versions),**
- **** data only available in GEN2 version 2.**

PRT_007 Printouts shall use the following data blocks and/or data records, in accordance with the following meanings and formats:

Block or record number Meaning	Data Format
1 <i>Date and time at which the document is printed.</i>	▼ dd/mm/yyyy hh:mm (UTC)
2 <i>Type of printout.</i> Block identifier VU generation and version** Printout pictogram combination (see App. 3), Speed limiting device setting (Over speeding printout only)	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> -----▼----- GEN2 v2 </div> Picto xxx km/h
3 <i>Card holder identification.</i> Block identifier. P= people pictogram Card holder surname Card holder first name(s) (if any) Card identification Card expiry date (if any) and Card generation number (GEN-1 or GEN-2)* and version**	<div style="border: 1px solid black; padding: 5px;"> -----P----- P Last_Name_____ First_Name_____ Card_Identification____ dd/mm/yyyy - GEN-2 v2 </div>

In the case where the card is a non-personal card, and holds no card holder surname, the company or workshop or control body name shall be printed instead.

~~* The card generation number can only be printed by smart tachograph.~~

4 <i>Vehicle identification.</i> Block identifier VIN Registering Member State Contracting Party and VRN	<div style="border: 1px solid black; padding: 5px;"> -----A----- A VIN_____ Nat/VRN_____ </div>
4a <i>Vehicle by-default load type**</i> pi = by-default load type of the vehicle pictogram**	pi
5 <i>VU identification.</i> Block identifier VU manufacturer's name VU part number VU generation number*	<div style="border: 1px solid black; padding: 5px;"> -----B----- B VU_Manufacturer_____ VU_Part_Number__ GEN-2 </div>

~~* The card generation number can only be printed by smart tachograph.~~

6 <i>Last calibration of the tachograph</i> Block identifier Workshop name Workshop card identification Date of the calibration	<div style="border: 1px solid black; padding: 5px;"> -----T----- T Last_Name_____ Card_Identification____ T dd/mm/yyyy </div>
7 <i>Last control (by a control officer)</i>	

Block identifier	-----□-----
Controller's card identification	Card_Identification_____
Control date, time and type	□ dd/mm/yyyy hh:mm ppppp
Type of the control: Up to five pictograms. The type of control can be (a combination) of:	
■: Card downloading, ▼: VU downloading, ▽: printing, □: Displaying, T: Roadside calibration checking	
8 Driver activities stored on a card in order of occurrence	
Block identifier	-----□-----
Enquiry date (calendar day subject of the printout) + Daily card presence counter	dd/mm/yyyy xxx
8a Out of scope condition in the beginning of this day (leave blank if no out of scope condition open)	-----OUT-----
8b Load type in the beginning of this day** (if the card is inserted in a VU, leave blank otherwise), pi=load type pictogram**	-----pi-----
8.1 Period during which the card was not inserted	
8.1a Record identifier (start of period)	-----
8.1b Unknown period. Start time, duration	? hh:mm hhhmm
8.1c Activity manually entered. Activity pictogram, start time, duration	A hh:mm hhhmm
8.2 Card insertion in slot S	
Record identifier; S = Slot pictogram	-----S-----
Vehicle registering Member State Contracting Party and VRN	■ Nat/VRN_____
Vehicle odometer at card insertion	x xxx xxx km
pi = vehicle load type at card insertion**	pi
8.3 Activity (while card was inserted)	
Activity pictogram, start time, duration, crew status (crew pictogram if CREW, blanks if SINGLE).	A hh:mm hhhmm □□
8.3a Specific condition. Time of entry, specific condition pictogram (or pictogram combination).	hh:mm ---pppp---
8.4 Card withdrawal	
Vehicle odometer and distance travelled since last insertion for which odometer is known	x xxx xxx km; x xxx km
9 Driver activities stored in a VU per slot in chronological order	
Block identifier	-----□-----
Enquiry date (calendar day subject of the printout)	dd/mm/yyyy
Vehicle odometer at 00:00 and 24:00	x xxx xxx - x xxx xxx km
10 Activities carried in slot S	
Block identifier	-----S-----
10a Out of scope condition in the beginning of this day (leave blank if no out of scope condition open)	-----OUT-----
10.1 Period where no card is inserted in slot S	
Record identifier.	-----
No Card inserted	□□---
Vehicle odometer at beginning of period	x xxx xxx km
10.2 Card insertion	
Card insertion Record identifier	-----
Driver's name	□ Last_Name_____
Driver's first name	First_Name_____

	Driver's Card identification Card expiry date (if any) and Card generation number (GEN-1 or GEN-2)* and version** Registering MS Contracting Party and VRN of previous vehicle used Date and time of card withdrawal from previous vehicle Blank line Vehicle odometer at card insertion, Manual entry of driver activities flag (M if yes, Blank if No). If no card insertion of a driver card happened on the day for which the printout is done then for block 10.2 the odometer data reading from the last available card insertion before that day shall be used.	Card_Identification_____ dd/mm/yyyy - GEN-2 v2 A+Nat/VRN_____ dd/mm/yyyy hh:mm x xxx xxx km M
10.3	Activity Activity pictogram, start time, duration, crew Status (crew pictogram if CREW, blanks if SINGLE).	A hh:mm hh h mm ⓐ
10.3a	<i>Specific condition.</i> Time of entry, specific condition pictogram (or pictogram combination).	hh:mm ---pppp---
10.4	<i>Card withdrawal or End of 'No Card' period</i> Vehicle odometer at card withdrawal or at end of 'no card' period and distance travelled since insertion, or since beginning of the 'No Card' period.	x xxx xxx km ; x xxx km
* The card generation number can only be printed by smart tachograph.		
11	Daily summary Block identifier	-----Σ-----
11.1	VU summary of periods without card in driver slot Block identifier	1ⓐ---
11.2	VU summary of periods without card in co-driver slot Block identifier	2ⓐ---
11.3	VU daily summary per driver Record identifier Driver's surname Driver's first name(s) Driver's card identification	----- ⓐ Last_Name_____ First_Name_____ Card_Identification_____
11.4	Entry of place where a daily work period begins and/or ends pi=location begin / end pictogram, time, country, region, latitude of the recorded position*, authentication status** longitude of the recorded position*, authentication status** timestamp when position was determined*, authentication status** Odometer	pihh:mm Cou Reg lat ±DDD°MM.M' ⓐ lon ±DDD°MM.M' ⓐ hh+mm dd/mm/yyyy hh:mm ⓐ x xxx xxx km
11.5	Entry of place where a daily work period begins and/or ends Positions after 3 hours accumulated driving time*	

11.5a	<p>Border Crossing**</p> <p>pi=position after 3 hours continuous accumulated driving time*, time of the record *</p> <p>longitude of the recorded position*, authentication status**</p> <p>latitude of the recorded position*, authentication status**</p> <p>timestamp when position was determined*, authentication status**</p> <p>Odometer*</p>	<pre> pihh:mm lat ±DDD°MM.M' lon ±DDD°MM.M' dd/mm/yyyy hh:mm x xxx xxx km </pre>
-------	---	--

11.5a	<p>Border Crossing**</p> <p>pi=position where the vehicle has crossed the border of a country**</p> <p>Country that the vehicle was leaving/entering**</p> <p>latitude of the recorded position**, authentication status**</p> <p>longitude of the recorded position**, authentication status**</p> <p>timestamp when position was determined**, authentication status**</p> <p>Odometer**</p>	<pre> pi Cou + Cou lat ± DD°MM.M' lon ±DDD°MM.M' dd/mm/yyyy hh:mm x xxx xxx km </pre>
-------	---	---

11.5b	<p>Load/unload operation**</p> <p>pi=position where a load/unload operation has occurred, time of the record**</p> <p>latitude of the recorded position**, authentication status**</p> <p>longitude of the recorded position**, authentication status**</p> <p>timestamp when position was determined**</p> <p>Odometer**</p>	<pre> pihh:mm lat ± DD°MM.M' lon ±DDD°MM.M' dd/mm/yyyy hh:mm x xxx xxx km </pre>
-------	--	--

11.6	<p>Activity totals (from a card)</p> <p>Total driving duration, distance travelled</p> <p>Total working and availability duration</p> <p>Total resting and unknown duration</p> <p>Total duration of crew activities</p>	<pre> ⊙ hhhmm x xxx km * hhhmm ☐ hhhmm ┌ hhhmm ? hhhmm ⊙⊙ hhhmm </pre>
------	---	--

11.7	<p>Activity totals (periods without card driver slot)</p> <p>Total driving duration, distance travelled</p> <p>Total working and availability duration</p> <p>Total resting duration</p>	<pre> ⊙ hhhmm x xxx km * hhhmm ☐ hhhmm ┌ hhhmm </pre>
------	---	---

11.8	<p>Activity totals (periods without card co-driver slot)</p> <p>Total working and availability duration</p> <p>Total resting duration</p>	<pre> * hhhmm ☐ hhhmm ┌ hhhmm </pre>
------	--	--

11.9	<p>Activity totals (per driver both slots included)</p> <p>Total driving duration, distance travelled</p> <p>Total working and availability duration</p> <p>Total resting duration</p> <p>Total duration of crew activities</p>	<pre> ⊙ hhhmm x xxx km * hhhmm ☐ hhhmm ┌ hhhmm ⊙⊙ hhhmm </pre>
------	--	--

When a daily printout is required for the current day, daily summary information is computed with available data at the time of the printout.

12	<p>Events and/or faults stored on a card</p> <p>12.1 Block identifier last 5 'Events and Faults' from a card</p>	<pre> -----!☒----- </pre>
----	---	---

12.2	Block identifier all recorded 'Events' on a card	-----!■-----
12.3	Block identifier all recorded 'Faults' on a card	-----*■-----
12.4	<i>Event and/or Fault record</i> Record identifier Event/fault pictogram, record purpose, date time of start, Additional event/fault code (if any), duration Registering Member State Contracting Party & VRN of vehicle in which the event or fault occurred	----- Pic (p) dd/mm/yyyy hh:mm !xx hhmm ■ Nat/VRN _____
13	Events and/or faults stored or on-going in a VU	
13.1	Block identifier last 5 'Events and Faults' from VU	-----!*■-----
13.2	Block identifier all recorded or on-going 'Events' in a VU	-----!■-----
13.3	Block identifier all recorded or on-going 'Faults' in a VU	-----*■-----
13.4	<i>Event and/or fault record</i> Record identifier Event/fault pictogram, record purpose, date time of start, Additional event/fault code (if any), No of similar events this day, duration Identification of the cards inserted at start or end of the event or fault (up to 4 lines without repeating twice the same card numbers) Case where no card was inserted Manufacturer specific data The record purpose (p) is a numerical code explaining why the event or fault was recorded, coded in accordance with the data element EventFaultRecordPurpose. The Literal is a tachograph manufacturer specific literal with 12 characters maximum. The ErrorCode is a tachograph manufacturer specific error code with 12 characters maximum.	----- Pic (p) dd/mm/yyyy hh:mm !xx (xxx) hhmm Card_Identification_____ Card_Identification_____ Card_Identification_____ Card_Identification_____ ■--- < Literal><ErrorCode>
14	VU Identification Block identifier VU manufacturer name VU manufacturer address VU part number VU approval number VU serial number VU year of manufacture VU generation and version** VU software version and installation date Version of the stored digital map**	-----■----- ■ Name _____ Address _____ PartNumber _____ Apprv _____ S/N _____ ¥yyyy GEN2 v2 V xxxx dd/mm/yyyy ■ XXXXXXXXXXXXXX

15	Sensor identification	Block identifier	-----1-----
15.1	Pairing record	Sensor serial number (S/N = serialNumber in decimal, MY = monthYear in decimal, T = type in decimal, MC = manufacturerCode in hexadecimal, see Sub-appendix 1, ExtendedSerialNumber)	1 S/N _____ MY__ T__ MC_
		Sensor approval number	Apprv _____
		Sensor pairing date	dd/mm/yyyy hh:mm

16	GNSS identification*	Block identifier*	-----1-----
16.1	Coupling record*	External GNSS facility serial number* (S/N = serialNumber in decimal, MY = monthYear in decimal, T = type in decimal, MC = manufacturerCode in hexadecimal, see Appendix 1, ExtendedSerialNumber)	1 S/N _____ MY__ T__ MC_
		External GNSS facility approval number*	Apprv _____
		External GNSS facility coupling date*	dd/mm/yyyy hh:mm

The calibration purpose (p) is a numerical code explaining why these calibration parameters were recorded, coded in accordance with the data element CalibrationPurpose.

16a **Remote communication facility identification****
Block identifier**

```
-----T-----
```

16a.1 **Remote communication facility serial number****
Remote communication facility serial number** (S/N = **serialNumber** in decimal, MY = **monthYear** in decimal, T = **type** in decimal, MC = **manufacturerCode** in hexadecimal, see Appendix 1, **ExtendedSerialNumber**)

```
T S/N _____ MY_ T_ MC_
```

17 **Calibration data**

Block identifier

```
-----T-----
```

17.1 **Calibration record**

Record identifier
 Workshop having performed the calibration
 Workshop address
 Workshop card identification
 Workshop card expiry date
 Blank line
 Calibration date **time (oldTimeValue in the calibration record) + calibration purpose in hexadecimal**
 VIN
 Registering ~~Member State~~ **Contracting Party & VRN**
 Characteristic coefficient of vehicle
 Constant of the ~~recording equipment~~ **control device**
 Effective circumference of wheel tyres
 Size of tyres mounted
 Speed limiting device setting
 Old and new odometer values
pi=by-default load type of the vehicle**
Country in which the calibration has been performed and date time
Seal data (up to 5 seal records, 1 line for each used seal), ET = equipmentType in decimal, MC = manufacturerCode as two characters**, SI = sealIdentifier as 8 characters**, see Appendix 1, SealRecord)**

```
-----
T Workshop_name _____
  Workshop_address _____
Card_Identification _____
  dd/mm/yyyy

T dd/mm/yyyy hh:mm (p)

A VIN _____
  Nat/VRN _____

w xx xxx Imp/km
k xx xxx Imp/km
l xx xxx mm
* TyreSize _____
> xxx km/h
x xxx xxx - x xxx xxx km
pi
Cou dd/mm/yyyy hh:mm

[ET_ MC SI _____
```

The calibration purpose (p) is a numerical code explaining why these calibration parameters were recorded, coded in accordance with the data element CalibrationPurpose.

18 **Time adjustment**

Block identifier

```
-----@-----
```

18.1 **Time adjustment record**

Record identifier
 Old date and time
 New date and time
 Workshop having performed the time adjustment
 Workshop address
 Workshop card identification
 Workshop card expiry date

```
-----
!@ dd/mm/yyyy hh:mm
 @ dd/mm/yyyy hh:mm
T Workshop_name _____
  Workshop_address _____
Card_Identification _____
  dd/mm/yyyy
```

19 **Most recent event and Fault recorded in the VU**

	Block identifier	-----!x#-----
	Most recent event date time	! dd/mm/yyyy hh:mm
	Most recent fault date time	x dd/mm/yyyy hh:mm
20	Over speeding control information	
	Block identifier	----->>-----
	Date and time of last OVER SPEEDING CONTROL	>#dd/mm/yyyy hh:mm
	Date/time of first over speeding and number of over speeding events since	>>dd/mm/yyyy hh:mm (nnn)
21	Over speeding record	
21.1	Block identifier 'First over speeding after the last calibration'	----->>T-----
21.2	Block identifier 'The 5 most serious over the last 365 days'	----->>(365)-----
21.3	Block identifier 'The most serious for each of the last 10 days of occurrence'	----->>(10)-----
21.4	Record identifier	-----
	Date time and duration	>>dd/mm/yyyy hh:mm hhmm
	Max and average speeds, No. of similar events this day	xxx km/h xxx km/h (xxx)
	Driver's surname	# Last_Name_____
	Driver's first name(s)	First_Name_____
	Driver card identification	Card Identification_____
21.5	If no over speeding record exists in a block	>>---
22	Hand-written information	
	Block identifier	-----
22.1	Control Place	#*
22.2	Controller's signature	#
22.3	From time	#+
22.4	To time	+#
22.5	Driver's signature	#
	'Hand-written information'; Insert enough blank lines above a hand-written item, to be able to actually write the required information or to put a signature.	
23	Most recent cards inserted in VU*	
	Block identifier*	-----# ##-----
23.1	Inserted Card*	
	Record identifier*	----
	Type of card, Generation, Version, Manufacturer* ¹	T <gen> <version> <MC>
	Card Identification*	Card Identification
	Card Serial Number*	Card Serial Number
	Date and time of last card insertion*	dd/mm/yyyy hh:mm

¹ (everything in one line)

with

type of card: Pictogram, one character + space

gen: GEN1 or GEN2, 4 characters + space

version: up to 10 characters

MC: manufacturer code, 3 characters

3. Printout specifications

In this chapter the following notation conventions have been used:

N

 Print block or record number N

N

 Print block or record number N repeated as many times as necessary

X / Y

Print blocks or records X and/or Y as needed, and repeating as many times as necessary.

3.1 Driver Activities from Card Daily Printout

PRT_008 The driver activities from card daily printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Controller identification (if a control card is inserted in the VU)
3	Driver identification (from card subject of the printout + GEN)
4	Vehicle identification (vehicle from which printout is taken)
5	VU identification (VU from which printout is taken + GEN)
6	Last calibration of this VU
7	Last control the inspected driver has been subject to
8	Driver activities delimiter
8a	Out of scope condition in the beginning of this day
8b	Load type in the beginning of the day (if the card is inserted in a VU)
8.1a / 8.1b / 8.1c / 8.2 / 8.3 / 8.3a / 8.4	Activities of the driver in order of occurrence
11	Daily summary delimiter
11.4	Places entered in chronological order
11.5	GNSS data Positions after 3 hours accumulated driving time in chronological order
11.5a	Border crossings, in chronological order
11.5b	Load/unload operations, in chronological order
11.6	Activity totals
12.1	Events or faults from card delimiter
12.4	Event/Fault records (Last 5 events or faults stored in the card)
13.1	Events or faults from VU delimiter
13.4	Event/Fault records (Last 5 events or faults stored or on-going in the VU)
22.1	Control place
22.2	Controller's signature
22.5	Driver's signature

3.2. Driver Activities from VU Daily Printout

PRT_009 The driver activities from VU daily printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Card holder identification (for all cards inserted in VU + GEN)
4	Vehicle identification (vehicle from which printout is taken)
4a	Vehicle by-default load type
5	VU identification (VU from which printout is taken + GEN)
6	Last calibration of this VU
7	Last control on this tachograph
9	Driver activities delimiter
10	Driver slot delimiter (slot 1)

10a	Out of scope condition in the beginning of this day
10.1 / 10.2 / 10.3 / 10.3a / 10.4	Activities in chronological order (driver slot)
10	Co-driver slot delimiter (slot 2)
10a	Out of scope condition in the beginning of this day
10.1 / 10.2 / 10.3 / 10.3a / 10.4	Activities in chronological order (co-driver slot)
11	Daily summary delimiter
11.1	Summary of periods without card in driver slot
11.4	Places entered in chronological order
11.5	GNSS data Positions after 3 hours accumulated driving time in chronological order
11.5a	Border crossings, in chronological order
11.5b	Load/unload operations, in chronological order
11.67	Activity totals
11.2	Summary of periods without card in co-driver slot
11.4	Places entered in chronological order
11.5	GNSS data Positions after 3 hours accumulated driving time in chronological order
11.5a	Border crossings, in chronological order
11.5b	Positions where load/unload operation has occurred, in chronological order
11.8	Activity totals
11.3	Summary of activities for a driver both slots included
11.4	Places entered by this driver in chronological order
11.5	GNSS data Positions after 3 hours accumulated driving time in chronological order
11.5a	Border crossings, in chronological order
11.5b	Load/unload operations, in chronological order
11.9	Activity totals for this driver
13.1	Events faults delimiter
13.4	Event/Fault records (Last 5 events or faults stored or on-going in the VU)
22.1	Control place
22.2	Controller's signature
22.3	From time (space available for a driver without a card to indicate
22.4	To time which periods are relevant to himself)
22.5	Driver's signature

3.3 Events and Faults from Card Printout

PRT_010 The events and faults from card printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Controller identification (if a control card is inserted in the VU + GEN)
3	Driver identification (from card subject of the printout)
4	Vehicle identification (vehicle from which printout is taken)
12.2	Events delimiter
12.4	Event records (all events stored on the card)

12.3	Faults delimiter
12.4	Fault records (all faults stored on the card)
22.1	Control place
22.2	Controller's signature
22.5	Driver's signature

3.4. Events and Faults from VU Printout

PRT_011 The events and faults from VU printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Card holder identification (for all cards inserted in VU + GEN)
4	Vehicle identification (vehicle from which printout is taken)
13.2	Events delimiter
13.4	Event records (All Events stored or on-going in the VU)
13.3	Faults delimiter
13.4	Fault records (All Faults stored or on-going in the VU)
22.1	Control place
22.2	Controller's signature
22.5	Driver's signature

3.5 Technical data Printout

PRT_012 The technical data printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Card holder identification (for all cards inserted in VU + GEN)
4	Vehicle identification (vehicle from which printout is taken)
14	VU identification
15	Sensor identification
15.1	Sensor Pairing data (all data available in chronological order)
16	GNSS identification
16.1	External GNSS facility coupling data (all data available in chronological order)
16a	Remote communication facility identification
16a.1	Remote communication facility serial number
17	Calibration data delimiter
17.1	Calibration records (all records available in chronological order)
18	Time adjustment delimiter
18.1	Time adjustment records (all records available from time adjustment and from calibration data records)
19	Most recent event and Fault recorded in the VU
2	Type of printout (indicates the end of the printout)

3.6 Over speeding Printout

PRT_013 The over speeding printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Card holder identification (for all cards inserted in VU + GEN)
4	Vehicle identification (vehicle from which printout is taken)

20	Over speeding control information
21.1	Over speeding data identifier
21.4 / 21.5	First over speeding after the last calibration
21.2	Over speeding data identifier
21.4 / 21.5	The 5 most serious over speeding events over the last 365 days
21.3	Over speeding data identifier
21.4 / 21.5	The most serious over speeding for each of the last 10 days of occurrence
22.1	Control place
22.2	Controller's signature
22.5	Driver's signature

3.7 Historic of inserted cards

PRT_014 The historic of inserted cards printout shall be in accordance with the following format:

1	Date and time at which the document is printed
2	Type of printout
3	Card holder identifications (of all cards inserted in the VU)
23	Most recent card inserted in the VU
23.1	Inserted cards (up to 88 records)
12.3	Type of printout (indicates the end of the printout) Faults delimiter

Sub-appendix 5. Display

In this sub-appendix the following format notation conventions have been used:

- characters printed in **bold** denote plain text to be displayed (display remains in normal character),
- normal characters denote variables (pictograms or data) to be replaced by their values for displaying:
- dd mm yyyy: day, month, year,
- hh: hours,
- mm: minutes,
- D: duration pictogram,
- EF: event or fault pictograms combination,
- O: mode of operation pictogram.

- DIS_001 The tachograph shall display data using the following formats:

Data	Format
Default display	
Local time	hh:mm
Mode of operation	O
Information related to the driver	1Dhh h mm h hhmm
Information related to the co-driver	2Dhh h mm
Out of scope condition opened	OUT
Warning display	
Exceeding continuous driving time	1⊙hh h mm h hhmm
Event or fault	EF
Other displays	
UTC date time	UTC ⊙dd/mm/yyyy or UTC ⊙dd.mm.yyyy hh:mm
Driver's continuous driving time and emulative accumulative break time	1⊙hh h mm h hhmm
Co-driver's continuous driving time and emulative accumulative break time	2⊙hh h mm h hhmm
Driver's emulated accumulated driving time for the previous and the current week	1⊙ hhh h mm

Co-driver's emulated accumulated driving time for the previous and the current week	20 hhh:mm
---	-------------

Sub-appendix 6. Front connector for calibration and download

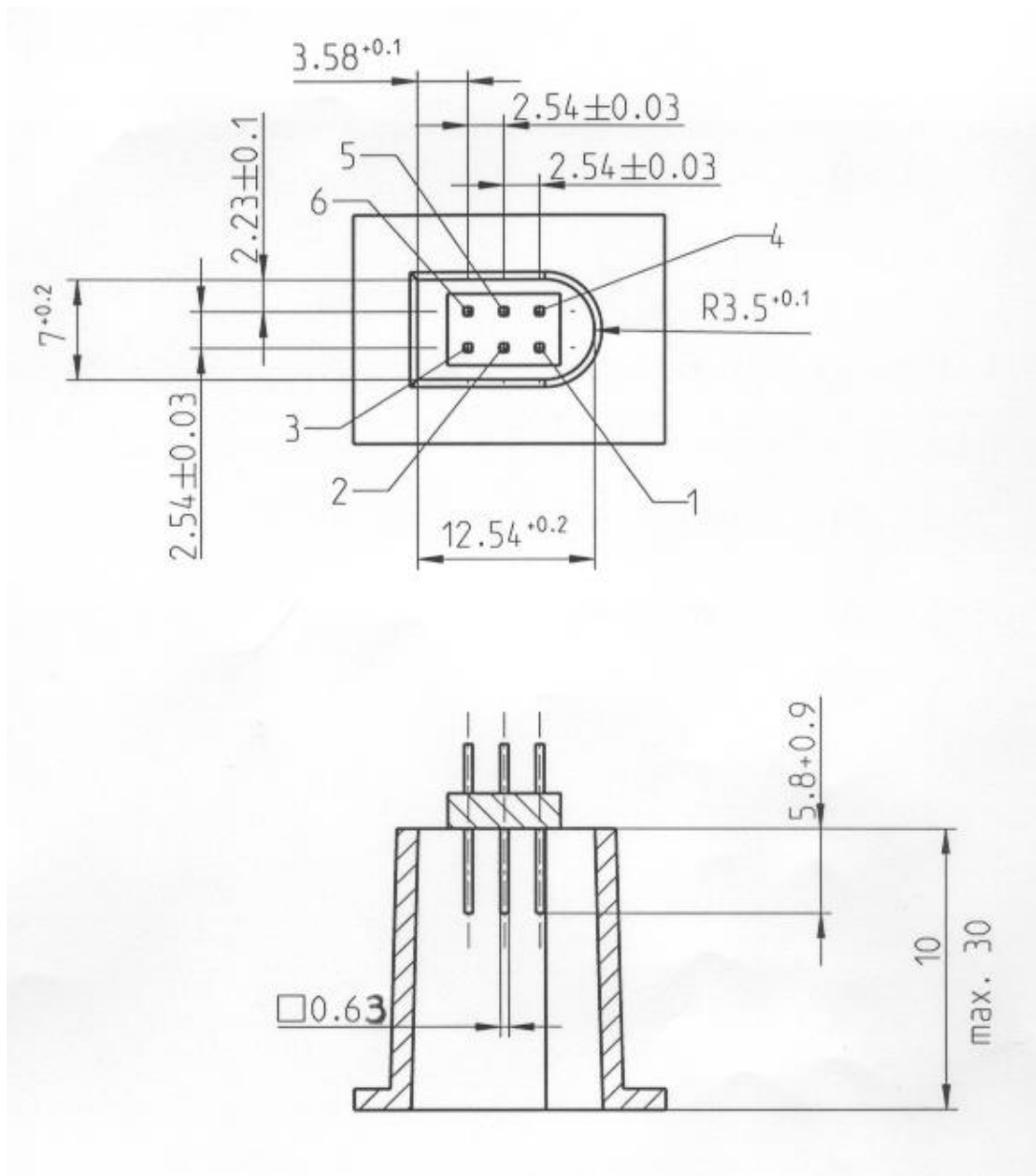
TABLE OF CONTENT

1. Hardware.....	329
1.1. Connector.....	329
1.2. Contact allocation	331
1.3. Block diagram	331
2. Downloading interface.....	331
3. Calibration interface.....	332

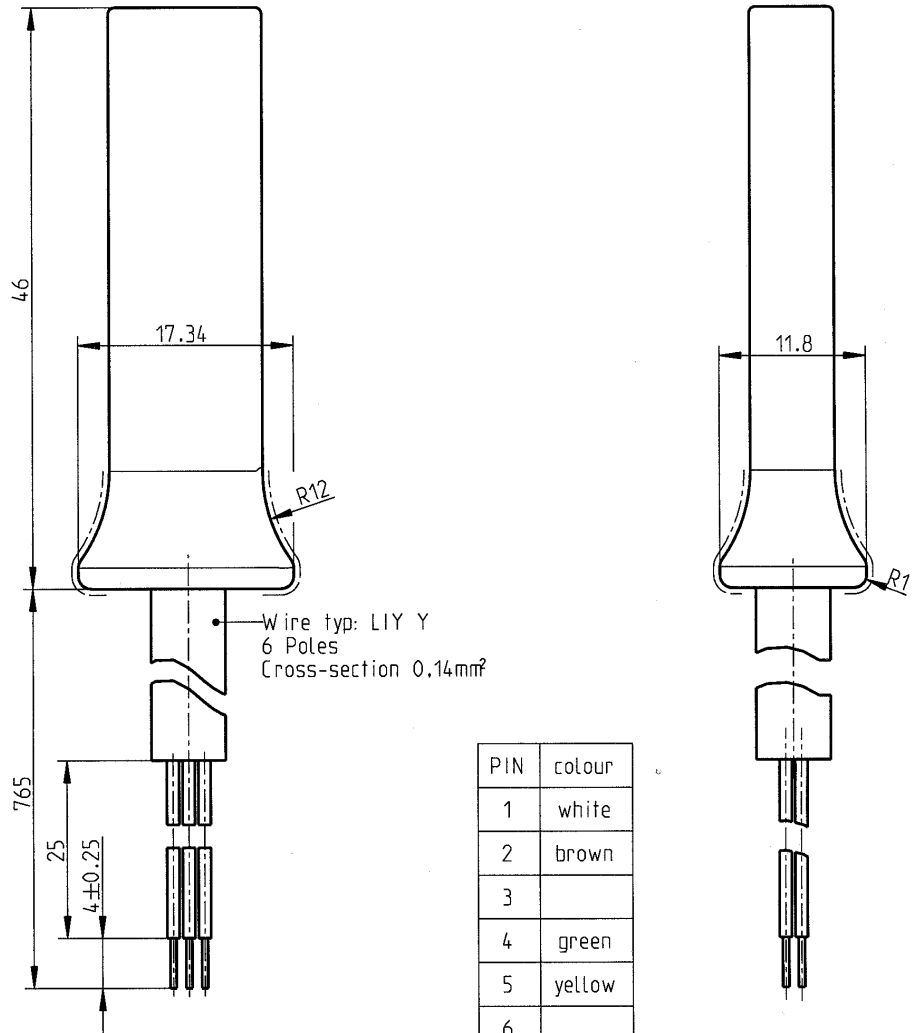
1. Hardware

1.1. Connector

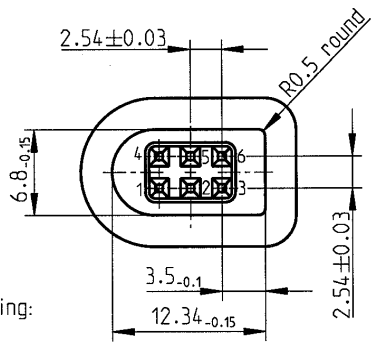
INT_001 The downloading/calibration connector shall be a 6 pin connector, accessible on the front panel without the need to disconnect any part of the tachograph, and shall comply with the following drawing (all dimensions in millimetres):



The following diagram shows a typical 6 pin mating plug:



PIN	colour
1	white
2	brown
3	
4	green
5	yellow
6	



Contact:
47 745-001
Socket housing:
650 43-034

--- Surface pattern
VDI 3400 Ref. 36 Ra 6.3 μm

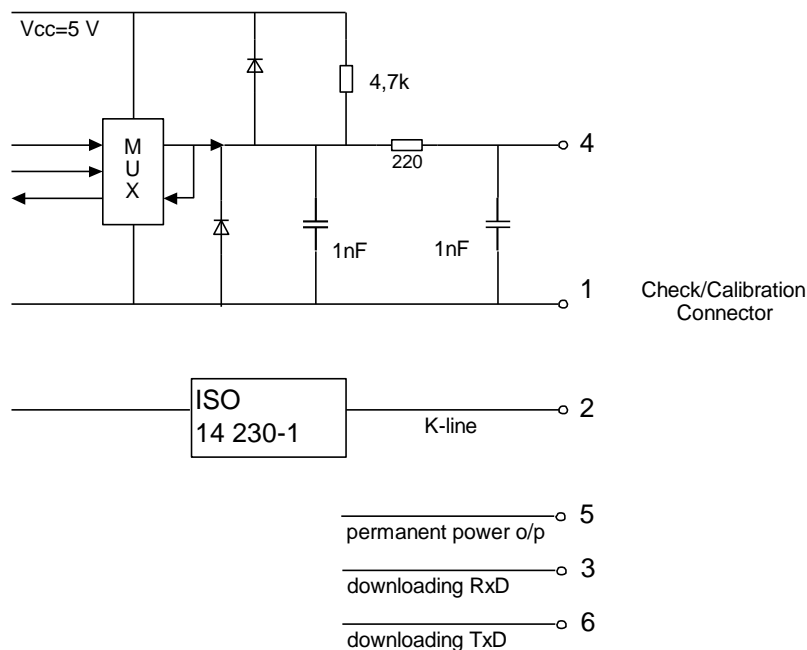
1.2. Contact allocation

INT_002 Contacts shall be allocated in accordance with the following table:

Pin	Description	Remark
1	Battery minus	Connected to the battery minus of the vehicle
2	Data communication	K-line (ISO 14230-1)
3	RxD – Downloading	Data input to tachograph
4	Input/output signal	Calibration
5	Permanent power output	The voltage range is specified to be that of the vehicle power minus 3V to allow for the voltage drop across the protective circuitry Output 40 mA
6	TxD – Downloading	Data output from tachograph

1.3. Block diagram

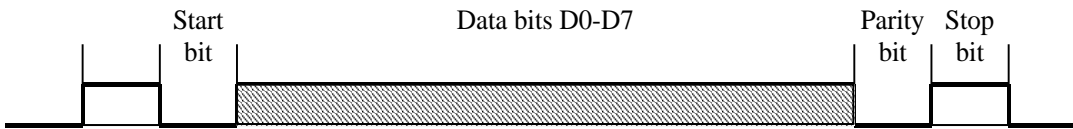
INT_003 The block diagram shall comply with the following:



2. Downloading interface

INT_004 The downloading interface shall comply to RS232 specifications.

INT_005 The downloading interface shall use one start bit, 8 data bits LSB first, one even parity bit and 1 stop bit.



Data byte organisation

- Start bit: one bit with logic level 0;
- Data bits: transmitted with LSB first;
- Parity bit: even parity
- Stop bit: one bit with logic level 1

When numerical data composed by more than one byte are transmitted, the most significant byte is transmitted first and the least significant byte last.

INT_006 Transmission baud rates shall be adjustable from 9 600 bps to 115 200 bps. Transmission shall be achieved at the highest possible transmission speed, the initial baud rate after a start of communication being set at 9 600 bps.

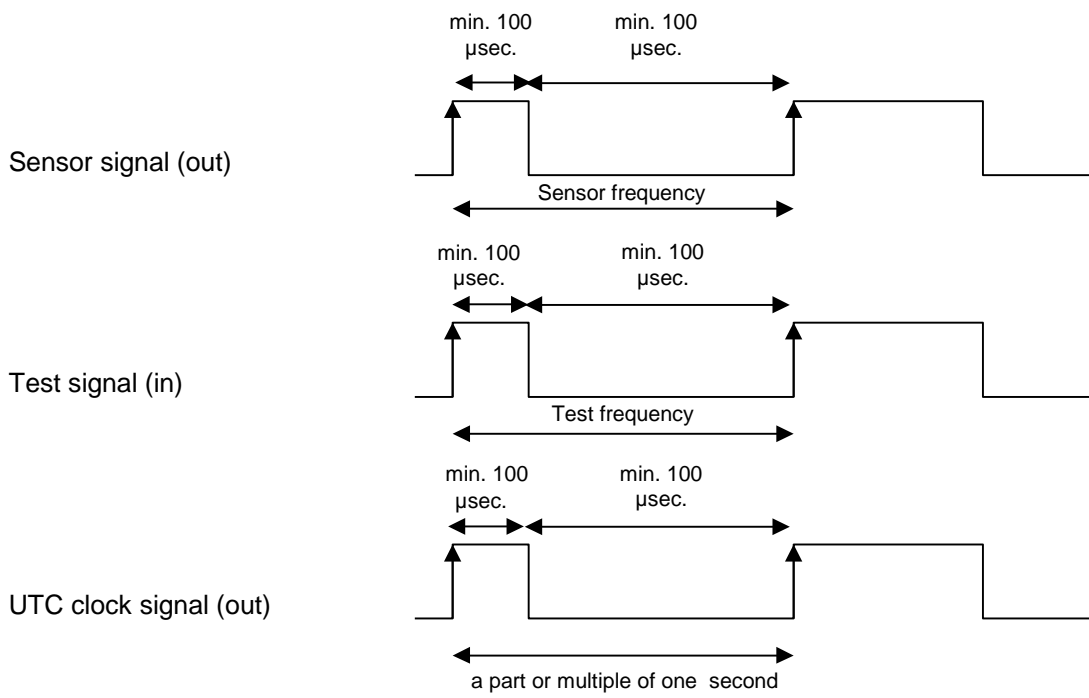
3. Calibration interface

INT_007 The data communication shall comply to ISO 14230-1 Road vehicles - Diagnostic systems - Keyword protocol 2000 - Part 1: Physical layer, First edition: 1999.

INT_008 The input/output signal shall comply with the following electrical specification:

Parameter	Minimum	Typical	Maximum	Remark
U _{low} (in)			1,0 V	I = 750 µA
U _{high} (in)	4 V			I = 200 µA
Frequency			4 kHz	
U _{low} (out)			1,0 V	I = 1 mA
U _{high} (out)	4 V			I = 1 mA

INT_009 The input/output signal shall comply with the following timing diagrams:



Sub-appendix 7. Data downloading protocols

TABLE OF CONTENT

1. Introduction.....	325
1.1. Scope.....	325
1.2. Acronyms and notations.....	325
2. V.U. data downloading.....	326
2.1. Download procedure	326
2.2. Data download protocol.....	327
2.2.1 Message structure	328
2.2.2 Message types	341
2.2.2.1 Start Communication Request (SID 81)	341
2.2.2.2 Positive Response Start Communication (SID C1)	341
2.2.2.3 Start Diagnostic Session Request (SID 10)	341
2.2.2.4 Positive Response Start Diagnostic (SID 50)	342
2.2.2.5 Link Control Service (SID 87)	342
2.2.2.6 Link Control Positive Response (SID C7).....	342
2.2.2.7 Request Upload (SID 35).....	342
2.2.2.8 Positive Response Request Upload (SID 75).....	342
2.2.2.9 Transfer Data Request (SID 36).....	342
2.2.2.10 Positive Response Transfer Data (SID 76).....	342
2.2.2.11 Request Transfer Exit (SID 37)	343
2.2.2.12 Positive Response Request Transfer Exit (SID 77).....	343
2.2.2.13 Stop Communication Request (SID 82)	343
2.2.2.14 Positive Response Stop Communication (SID C2)	343
2.2.2.15 Acknowledge Sub Message (SID 83).....	343
2.2.2.16 Negative Response (SID 7F)	344
2.2.3 Message flow	345
2.2.4 Timing	346
2.2.5 Error handling	346
2.2.5.1 Start Communication phase	346
2.2.5.2 Communication phase.....	349
2.2.6 Response Message content	350
2.2.6.1 Positive Response Transfer Data Download Interface Version	
2.2.6.2 Positive Response Transfer Data Overview.....	350
2.2.6.3 Positive Response Transfer Data Activities	351
2.2.6.4 Positive Response Transfer Data Events and Faults	355
2.2.6.5 Positive Response Transfer Data Detailed Speed	355
2.2.6.6 Positive Response Transfer Data Technical Data	356
2.3. ESM File storage.....	356
3. Tachograph cards downloading protocol.....	357
3.1. Scope.....	357
3.2. Definitions	357
3.3. Card Downloading.....	357
3.3.1 Initialisation sequence	358
3.3.2 Sequence for un-signed data files.....	359
3.3.3 Sequence for Signed data files.....	360
3.3.4 Sequence for resetting the calibration counter.....	360
3.4. Data storage format.....	360
3.4.1 Introduction	360
3.4.2 File format	361
4. Downloading a tachograph card via a vehicle unit.....	361

1. Introduction

This sub-appendix specifies the procedures to follow in order to perform the different types of data download to an External Storage Medium, together with the protocols that must be implemented to assure the correct data transfer and the full compatibility of the downloaded data format to allow any controller to inspect these data and be able to control their authenticity and their integrity before analysing them.

1.1. Scope

Data may be downloaded to an ESM:

- from a Vehicle Unit by an Intelligent Dedicated Equipment (IDE) connected to the VU,
- from a tachograph card by an IDE fitted with a card interface device (IFD),
- from a tachograph card via a vehicle unit by an IDE connected to the VU.

To give the possibility to verify the authenticity and integrity of downloaded data stored on an ESM, data is downloaded with a signature appended in accordance with ~~Appendix~~ **Sub-appendix 11** Common Security Mechanisms. The source equipment (VU or card) identification and its security certificates (~~Member State Contracting Party~~ and equipment) are also downloaded. The verifier of the data must possess independently a trusted ~~European~~ **Root** public key.

Data downloaded from a VU are signed using Sub-appendix 11 Common Security Mechanisms Part B (Second-generation tachograph system), except when drivers' control is performed by a non EU control authority, using a first generation control card, in which case data are signed using Sub-appendix 11 Common Security Mechanisms Part A (First-generation tachograph system), as requested by Sub-appendix 15 Migration, requirement MIG_015.

This sub-appendix specifies therefore two types of data downloads from the VU:

- Generation 2 type of VU data download, providing the generation 2 data structure, signed using Sub-appendix 11 Common Security Mechanisms Part B,
- Generation 1 type of VU data download, providing the generation 1 data structure, signed using Sub-appendix 11 Common Security Mechanisms Part A.

Similarly, there are two types of data downloads from second generation driver cards inserted in a VU, as specified in paragraphs 3 and 4 of this sub-appendix.

DDP_001 Data downloaded during one download session must be stored in the ESM within one single file.

1.2. Acronyms and notations

The following acronyms are used in this **sub**-appendix:

- AID** Application Identifier
- ATR** Answer To Reset
- CS** Checksum byte
- DF** Dedicated File
- DS_** Diagnostic Session

EF	Elementary File
ESM	External Storage Medium
FID	File Identifier (File ID)
FMT	Format Byte (first byte of message header)
ICC	Integrated Circuit Card
IDE	Intelligent Dedicated Equipment: The equipment used to perform data downloading to the ESM (e.g. Personal Computer)
IFD	Interface Device
KWP	Keyword Protocol 2000
LEN	Length Byte (last byte of message header)
PPS	Protocol Parameter Selection
PSO	Perform Security Operation
SID	Service Identifier
SRC	Source byte
TGT	Target Byte
TLV	Tag Length Value
TREP	Transfer Response Parameter
TRTP	Transfer Request Parameter
VU	Vehicle Unit

2. V.U. data downloading

2.1. Download procedure

- In order to carry on a VU data download, the operator must perform the following operations:
- Insert his tachograph card inside a card slot of the VU(*);
- Connect the IDE to the VU download connector;
- Establish the connection between the IDE and the VU;
- Select on the IDE the data to download and send the request to the VU;
- Close the download session.

(*) The card inserted will trigger the appropriate access rights to the downloading function and to the data. It shall, however, be possible to download data from a driver card inserted into one of the VU slots when no other card type is inserted in the other slot.

2.2. Data download protocol

The protocol is structured on a master-slave basis, with the IDE playing the master role and the VU playing the slave role.

The message structure, types and flow are principally based on the Keyword Protocol 2000 (KWP) (ISO 14230-2 Road vehicles – Diagnostic systems – Keyword protocol 2000 – Part2: Data link layer).

The application layer is principally based on the current draft to date of ISO 14229-1 (Road vehicles – Diagnostic systems – Part 1: Diagnostic services, version 6 of 22 February 2001).

2.2.1 Message structure

DDP_002 All the messages exchanged between the IDE and the VU are formatted with a structure consisting of three parts:

- Header composed by a Format byte (FMT), a Target byte (TGT), a Source byte (SRC) and possibly a Length byte (LEN),
- Data field composed by a Service Identifier byte (SID) and a variable number of data bytes, which can include an optional diagnostic session byte (DS_) or an optional transfer parameter byte (TRTP or TREP).
- Checksum composed by a Checksum byte (CS).
-

Header				Data field					Checksum
FMT	TGT	SRC	LEN	SID	DATA	CS
4 bytes				Max 255 bytes					1 byte

The TGT and SRC byte represent the physical address of the recipient and originator of the message. Values are F0 Hex for the IDE and EE Hex for the VU.

The LEN byte is the length of the Data field part.

The Checksum byte is the 8 bit sum series modulo 256 of all the bytes of the message excluding the CS itself.

FMT, SID, DS_, TRTP and TREP bytes are defined later in this document.

DDP_003 In the case where the data to be carried by the message is longer than the space available in the data field part, the message is actually sent in several sub messages. Each sub message bears a header, the same SID, TREP and a 2-byte sub message counter indicating the sub message number within the total message. To enable error checking and abort the IDE acknowledges every sub message. The IDE can accept the sub message, ask for it to be re-transmitted, request the VU to start again or abort the transmission.

DDP_004 If the last sub message contains exactly 255 bytes in the data field, a final sub message with an empty (except SID TREP and sub message counter) data field must be appended to show the end of the message.

Example:

Header	SID	TREP	Message			CS
4 Bytes	Longer than 255 Bytes					

Will be transmitted as:

Header	SID	TREP	00	01	Sub message 1	CS
4 Bytes	255 Bytes					

Header	SID	TREP	00	02	Sub message 2	CS
4 Bytes	255 Bytes					

...

Header	SID	TREP	xx	yy	Sub message n	CS
4 Bytes	Less than 255 Bytes					

or as:

Header	SID	TREP	00	01	Sub message 1	CS
4 Bytes	255 Bytes					

Header	SID	TREP	00	02	Sub message 2	CS
4 Bytes	255 Bytes					

...

Header	SID	TREP	xx	yy	Sub message n	CS
4 Bytes	255 Bytes					

Header	SID	TREP	xx	yy+1	CS
4 Bytes	4 bytes				

2.2.2 Message types

The communication protocol for data download between the VU and the IDE requires the exchange of 8 different message types.

The following table summarises these messages.

Message Structure	Max 4-Bytes Header				Max 255 Bytes Data			1-Byte CheckSum
	FMT	TGT	SRC	LEN	SID	DS_/TRTP	DATA	CS
IDE ->	←- VU							
Start Communication Request	81	EE	F0		81			E0
Positive Response Start Communication	80	F0	EE	03	C1		EA, 8F	9B
Start Diagnostic Session Request	80	EE	F0	02	10	81		F1
Positive Response Start Diagnostic	80	F0	EE	02	50	81		31
Link Control Service								
Verify Baud Rate (stage 1)								
9-600 Bd	80	EE	F0	04	87		01,01,01	EC
19-200 Bd	80	EE	F0	04	87		01,01,02	ED
38-400 Bd	80	EE	F0	04	87		01,01,03	EE
57-600 Bd	80	EE	F0	04	87		01,01,04	EF
115-200 Bd	80	EE	F0	04	87		01,01,05	F0
Positive Response Verify Baud Rate	80	F0	EE	02	C7		01	28
Transition Baud Rate (stage 2)	80	EE	F0	03	87		02,03	ED
Request Upload	80	EE	F0	0A	35		00,00,00,00; 00,FF,FF, FF,FF	99
Positive Response Request Upload	80	F0	EE	03	75		00,FF	D5
Transfer Data Request								
Overview	80	EE	F0	02	36	01 or 21		97
Activities	80	EE	F0	06	36	02 or 22	Date	CS
Events & Faults	80	EE	F0	02	36	03 or 23		99
Detailed Speed	80	EE	F0	02	36	04 or 24		9A
Technical Data	80	EE	F0	02	36	05 or 25		9B
Card download	80	EE	F0	02	36	06	Slot	CS
Positive Response Transfer Data	80	F0	EE	Len	76	TREP	Data	CS
Request Transfer Exit	80	EE	F0	01	37			96
Positive Response Request Transfer Exit	80	F0	EE	01	77			D6
Stop Communication Request	80	EE	F0	01	82			E1
Positive Response Stop Communication	80	F0	EE	01	C2			21
Acknowledge sub-message	80	EE	F0	Len	83		Data	CS
Negative responses								
General reject	80	F0	EE	03	7F	Sid-Req	10	CS
Service not supported	80	F0	EE	03	7F	Sid-Req	11	CS
Sub-function not supported	80	F0	EE	03	7F	Sid-Req	12	CS
Incorrect Message Length	80	F0	EE	03	7F	Sid-Req	13	CS
Conditions not correct or Request sequence error	80	F0	EE	03	7F	Sid-Req	22	CS
Request out of range	80	F0	EE	03	7F	Sid-Req	31	CS
Upload not accepted	80	F0	EE	03	7F	Sid-Req	50	CS
Response pending	80	F0	EE	03	7F	Sid-Req	78	CS
Data not available	80	F0	EE	03	7F	Sid-Req	FA	CS

Message Structure	Max 4 Bytes Header				Max 255 Bytes Data			1 Byte CheckSum
	FMT	TGT	SRC	LEN	SID	DS_ / TRTP	DATA	CS
IDE ->	<- VU							
Start Communication Request	81	EE	F0	02	81			E0
Positive Response Start Communication	80	F0	EE	03	C1		EA, 8F	9B
Start Diagnostic Session Request	80	EE	F0	02	10	81		F1
Positive Response Start Diagnostic	80	F0	EE	02	50	81		31
Link Control Service								
Verify Baud Rate (stage 1)								
9 600 Bd	80	EE	F0	04	87	01	01,01	EC
19 200 Bd	80	EE	F0	04	87	01	01,02	ED
38 400 Bd	80	EE	F0	04	87	01	01,03	EE
57 600 Bd	80	EE	F0	04	87	01	01,04	EF
115 200 Bd	80	EE	F0	04	87	01	01,05	F0
Positive Response Verify Baud Rate	80	F0	EE	02	C7	01		28
Transition Baud Rate (stage 2)	80	EE	F0	03	87	02	03	ED
Request Upload	80	EE	F0	0A	35		00,00,00,00, 00,FF,FF, FF,FF	99
Positive Response Request Upload	80	F0	EE	03	75		00,FF	D5
Transfer Data Request								
Download interface version	80	EE	F0	02	36	00		96
Overview	80	EE	F0	02	36	01, 21 or 31		CS
Activities	80	EE	F0	06	36	02, 22 or 32	Date	CS
Events & Faults	80	EE	F0	02	36	03, 23 or 33		CS
Detailed Speed	80	EE	F0	02	36	04 or 24		CS
Technical Data	80	EE	F0	02	36	05, 25 or 35		CS
Card download	80	EE	F0	02 or 03	36	06	Slot	CS
Positive Response Transfer Data	80	F0	EE	Len	76	TREP	Data	CS
Request Transfer Exit	80	EE	F0	01	37			96
Positive Response Request Transfer Exit	80	F0	EE	01	77			D6
Stop Communication Request	80	EE	F0	01	82			E1
Positive Response Stop Communication	80	F0	EE	01	C2			21
Acknowledge sub message	80	EE	F0	Len	83		Data	CS
Negative responses								
General reject	80	F0	EE	03	7F	Sid Req	10	CS
Service not supported	80	F0	EE	03	7F	Sid Req	11	CS
Sub function not supported	80	F0	EE	03	7F	Sid Req	12	CS
Incorrect Message Length	80	F0	EE	03	7F	Sid Req	13	CS
Conditions not correct or Request sequence error	80	F0	EE	03	7F	Sid Req	22	CS
Request out of range	80	F0	EE	03	7F	Sid Req	31	CS
Upload not accepted	80	F0	EE	03	7F	Sid Req	50	CS
Response pending	80	F0	EE	03	7F	Sid Req	78	CS
Data not available	80	F0	EE	03	7F	Sid Req	FA	CS

Notes:

- Sid Req = the Sid of the corresponding request.
- TREP = the TRTP of the corresponding request.
- Dark cells denote that nothing is transmitted.
- The term upload (as seen from the IDE) is used for compatibility with ISO 14229. It means the same as download (as seen from the VU).
- Potential 2-byte sub message counters are not shown in this table.
- Slot is the slot number, either "1" (card on driver slot) or "2" (card on co-driver slot)
- In case the slot is not specified, the VU shall select slot 1 if a card is inserted in this slot and it shall select slot 2 only in case it is specifically selected by the user.

- TRTP 24 is used for Generation 2, for version 1 and version 2 type of VU data download requests.
- TRTP 00, 31, 32, 33 and 35 are used for Generation 2 version 2 type of VU data download requests.
- TRTP 21, 22, 23, and 25 are used for Generation 2 version 1 type of VU data download requests.
- TRTP 01 to 05 are used for Generation 1 type of VU data download requests. They can optionally be accepted by Generation 2 type of VU, but only in the frame of drivers' control performed by a non-EU control authority, using a first generation control card.
- TRTP 11 to 1F are reserved for manufacturer specific download requests.
- ~~— TRTP 21 to 25 are used for Generation 2 type of VU data download requests, TRTP 01 to 05 are used for Generation 1 type of VU data download requests, which can only be accepted by the VU in the frame of drivers' control performed by non-European Union control authority, using a first generation control card,~~
- ~~TRTP 11 to 19 and 31 to 39 are reserved for manufacturer specific download requests.~~

2.2.2.1 Start Communication Request (SID 81)

DDP_005 This message is issued by the IDE to establish the communication link with the VU. Initial communications are always performed at 9600 baud (until baud rate is eventually changed using the appropriate Link control services).

2.2.2.2 Positive Response Start Communication (SID C1)

DDP_006 This message is issued by the VU to answer positively to a start communication request. It includes the 2 key bytes 'EA' '8F' indicating that the unit supports protocol with header including target source and length information.

2.2.2.3 Start Diagnostic Session Request (SID 10)

DDP_007 The Start Diagnostic Session request message is issued by the IDE in order to request a new diagnostic session with the VU. The sub function 'default session' (81 Hex) indicates a standard diagnostic session is to be opened.

2.2.2.4 Positive Response Start Diagnostic (SID 50)

DDP_008 The Positive Response Start Diagnostic message is sent by the VU to answer positively to Diagnostic Session Request.

2.2.2.5 Link Control Service (SID 87)

DDP_052 The Link Control Service is used by the IDE to initiate a change in baud rate. This takes place in two steps. In step one the IDE proposes the baud rate change, indicating the new rate. On receipt of a positive message from the VU the IDE sends out confirmation of the baud rate change to the VU (step two). The IDE then changes to the new baud rate. After receipt of the confirmation the VU changes to the new baud rate

2.2.2.6 Link Control Positive Response (SID C7)

DDP_053 The Link Control Positive Response is issued by the VU to answer positively to Link Control Service request (step one). Note that no response is given to the confirmation request (step two).

2.2.2.7 Request Upload (SID 35)

DDP_009 The Request Upload message is issued by the IDE to specify to the VU that a download operation is requested. To meet the requirements of **ISO 14229** data is included covering address, the size and format details for the data requested. As these are not known to the IDE prior to a download, the memory address is set to 0, format is unencrypted and uncompressed and the memory size is set to the maximum.

2.2.2.8 Positive Response Request Upload (SID 75)

DDP_010 The Positive Response Request Upload message is sent by the VU to indicate to the IDE that the VU is ready to download data. To meet the requirements of ISO 14229 data is included in this positive response message, indicating to the IDE that further Positive Response Transfer Data messages will include 00FF hex bytes maximum.

2.2.2.9 Transfer Data Request (SID 36)

DDP_011 The Transfer Data Request is sent by the IDE to specify to the VU the type of data that are to be downloaded. A one byte Transfer Request Parameter (TRTP) indicates the type of transfer.

There are ~~seven~~ types of data transfer. **For VU data download, two different TRTP values can be used for each transfer type:**

Data transfer type	TRTP value for generation 1 type of VU data download	TRTP value for generation 2, version 1 type of VU data download	TRTP value for generation 2, version 2 type of VU data download
Download interface version	Not used	Not used	00
Overview	01	21	31
Activities of a specified date	02	22	32
Events and faults	03	23	33
Detailed speed	04	24	24
Technical data	05	25	35

Data transfer type	TRTP value
Card download	06

DDP_054 It is mandatory for the IDE to request the overview data transfer (TRTP 01, **21** or ~~231~~) during a download session as this only will ensure that the VU certificates are recorded within the downloaded file (and allow for verification of digital signature).

In the ~~second~~ **third** case (TRTP 02, **22** or ~~232~~) the Transfer Data Request message includes the indication of the calendar day (TimeReal format) to be downloaded.

2.2.2.10 Positive Response Transfer Data (SID 76)

DDP_012 The Positive Response Transfer Data is sent by the VU in response to the Transfer Data Request. The message contains the requested data, with a Transfer Response Parameter (TREP) corresponding to the TRTP of the request.

DDP_055 In the first case (TREP 01, 21 **or 231**), the VU will send data helping the IDE operator to choose the data he wants to download further. The information contained within this message is:

- Security certificates,
- Vehicle identification,
- VU current date and time,
- Min and Max downloadable date (VU data),
- Indication of cards presence in the VU,
- Previous download to a company,
- Company locks,
- Previous controls.

2.2.2.11 Request Transfer Exit (SID 37)

DDP_013 The Request Transfer Exit message is sent by the IDE to inform the VU that the download session is terminated.

2.2.2.12 Positive Response Request Transfer Exit (SID 77)

DDP_014 The Positive Response Request Transfer Exit message is sent by the VU to acknowledge the Request Transfer Exit.

2.2.2.13 Stop Communication Request (SID 82)

DDP_015 The Stop Communication Request message is sent by the IDE to disconnect the communication link with the VU.

2.2.2.14 Positive Response Stop Communication (SID C2)

DDP_016 The Positive Response Stop Communication message is sent by the VU to acknowledge the Stop Communication Request.

2.2.2.15 Acknowledge Sub Message (SID 83)

DDP_017 The Acknowledge Sub Message is sent by the IDE to confirm receipt of each part of a message that is being transmitted as several sub messages. The data field contains the SID received from the VU and a 2-byte code as follows:

- MsgC +1 Acknowledges correct receipt of sub message number MsgC.
Request from the IDE to the VU to send next sub message
- MsgC indicates a problem with the receipt of sub message number MsgC.
Request from the IDE to the VU to send the sub message again.
- FFFF requests termination of the message.

This can be used by the IDE to end the transmission of the VU message for any reason.

The last sub message of a message (LEN byte < 255) may be acknowledged using any of these codes or not acknowledged.

The VU responses that will consist of several sub messages are:

- Positive Response Transfer Data (SID 76)

2.2.2.16 Negative Response (SID 7F)

DDP_018 The Negative Response message is sent by the VU in response to the above request messages when the VU cannot satisfy the request. The data fields of the message contains the SID of the response (7F), the SID of the request, and a code specifying the reason of the negative response. The following codes are available:

- 10 general reject

The action cannot be performed for a reason not covered below.

- 11 service not supported

The SID of the request is not understood.

- 12 sub function not supported

The DS_ or TRTP of the request is not understood, or there are no further sub messages to be transmitted.

- 13 incorrect message length

The length of the received message is wrong.

- 22 conditions not correct or request sequence error

The required service is not active or the sequence of request messages is not correct.

- 31 Request out of range

The request parameter record (data field) is not valid.

- 50 upload not accepted

The request cannot be performed (VU in a non appropriate mode of operation or internal fault of the VU).

- 78 response pending

The action requested cannot be completed in time and the VU is not ready to accept another request.

- FA data not available

The data object of a data transfer request are not available in the VU (e.g. no card is inserted, **generation 1 type of VU data download requested outside the frame of a driver's control by a non EU control authority ...**).

2.2.3 Message flow

A typical message flow during a normal data download procedure is the following:

IDE		VU
Start Communication Request	⇒ ⇐	Positive Response
Start Diagnostic Service Request	⇒ ⇐	Positive Response
Request Upload	⇒ ⇐	Positive Response
Transfer Data Request Overview	⇒ ⇐	Positive Response
Transfer Data Request #2	⇒ ⇐	Positive Response #1

Acknowledge Sub Message #1	⇄	
Acknowledge Sub Message #2	⇄	Positive Response #2
Acknowledge Sub Message #m	⇄	Positive Response #m
Acknowledge Sub Message (optional)	⇄	Positive Response (Data Field<255 Bytes)
...		
Transfer Data Request #n	⇄	Positive Response
Request Transfer Exit	⇄	Positive Response
Stop Communication Request	⇄	Positive Response

2.2.4 Timing

DDP_019 During normal operation the timing parameters shown in the following figure are relevant:

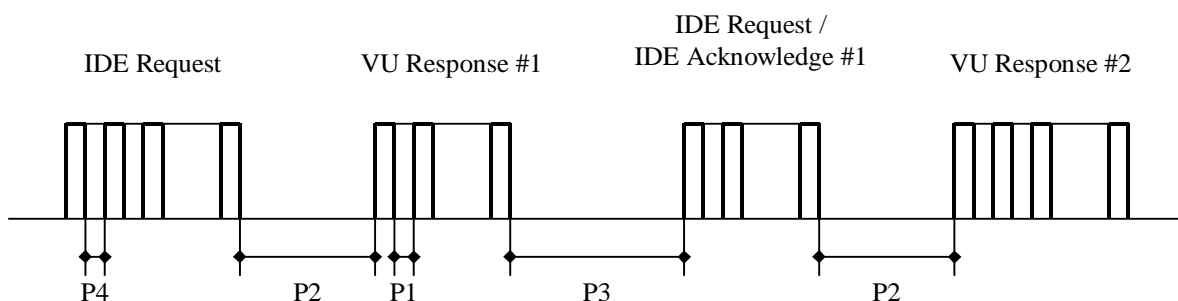


Figure 1 - Message flow, timing

Where:

P1 = Inter byte time for VU response.

P2 = Time between end of IDE request and start of VU response, or between end of IDE acknowledge and start of next VU response.

P3 = Time between end of VU response and start of new IDE request, or between end of VU response and start of IDE acknowledge, or between end of IDE request and start of new IDE request if VU fails to respond.

P4 = Inter byte time for IDE request.

P5 = Extended value of P3 for card downloading.

The allowed values for the timing parameters are showed in the following table (KWP extended timing parameters set, used in case of physical addressing for faster communication).

Timing Parameter	Lower limit Value (ms)	Upper limit value (ms)
P1	0	20
P2	20	1000 (*)
P3	10	5000
P4	5	20
P5	10	20 minutes

(*) if the VU responds with a Negative Response containing a code meaning “request correctly received, response pending”, this value is extended to the same upper limit value of P3.

2.2.5 Error handling

If an error occurs during the message exchange, the message flow scheme is modified depending on which equipment has detected the error and on the message generating the error.

In figure 2 and figure 3 the error handling procedures for the VU and the IDE are respectively shown.

2.2.5.1 Start Communication phase

DDP_020 If the IDE detects an error during the Start Communication phase, either by timing or by the bit stream, then it will wait for a period P3min before issuing again the request.

DDP_021 If the VU detects an error in the sequence coming from the IDE, it shall send no response and wait for another Start Communication Request message within a period P3 max.

2.2.5.2 Communication phase

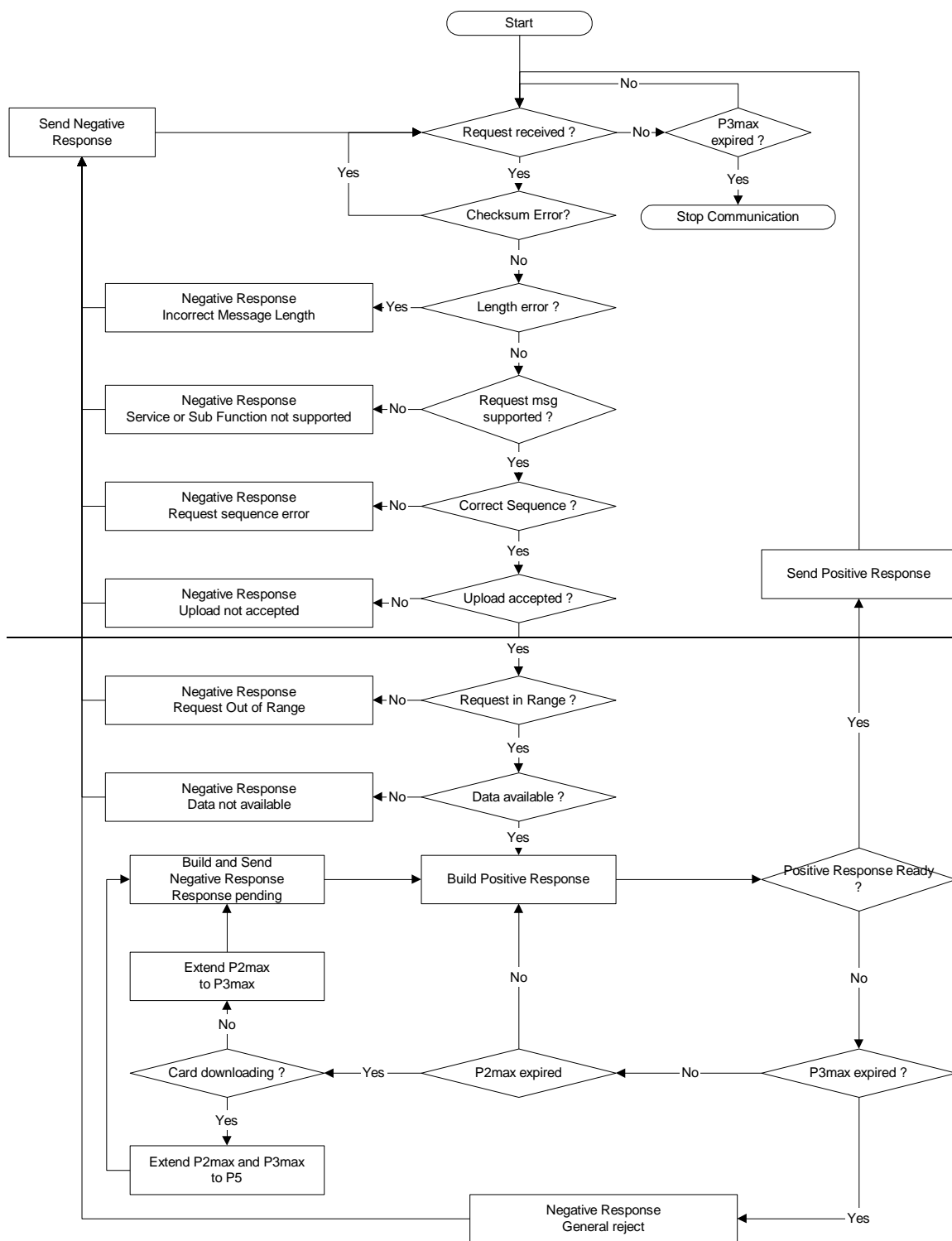
Two different error handling areas can be defined:

1. The VU detects an IDE transmission error.

DDP_022 For every received message the VU shall detect timing errors, byte format errors (e.g. start and stop bit violations) and frame errors (wrong number of bytes received, wrong checksum byte).

DDP_023 If the VU detects one of the above errors, then it sends no response and ignores the message received.

DDP_024 The VU may detect other errors in the format or content of the received message (e.g. message not supported) even if the message satisfies the length and checksum requirements; in such a case, the VU shall respond to the IDE with a Negative Response message specifying the nature of the error.



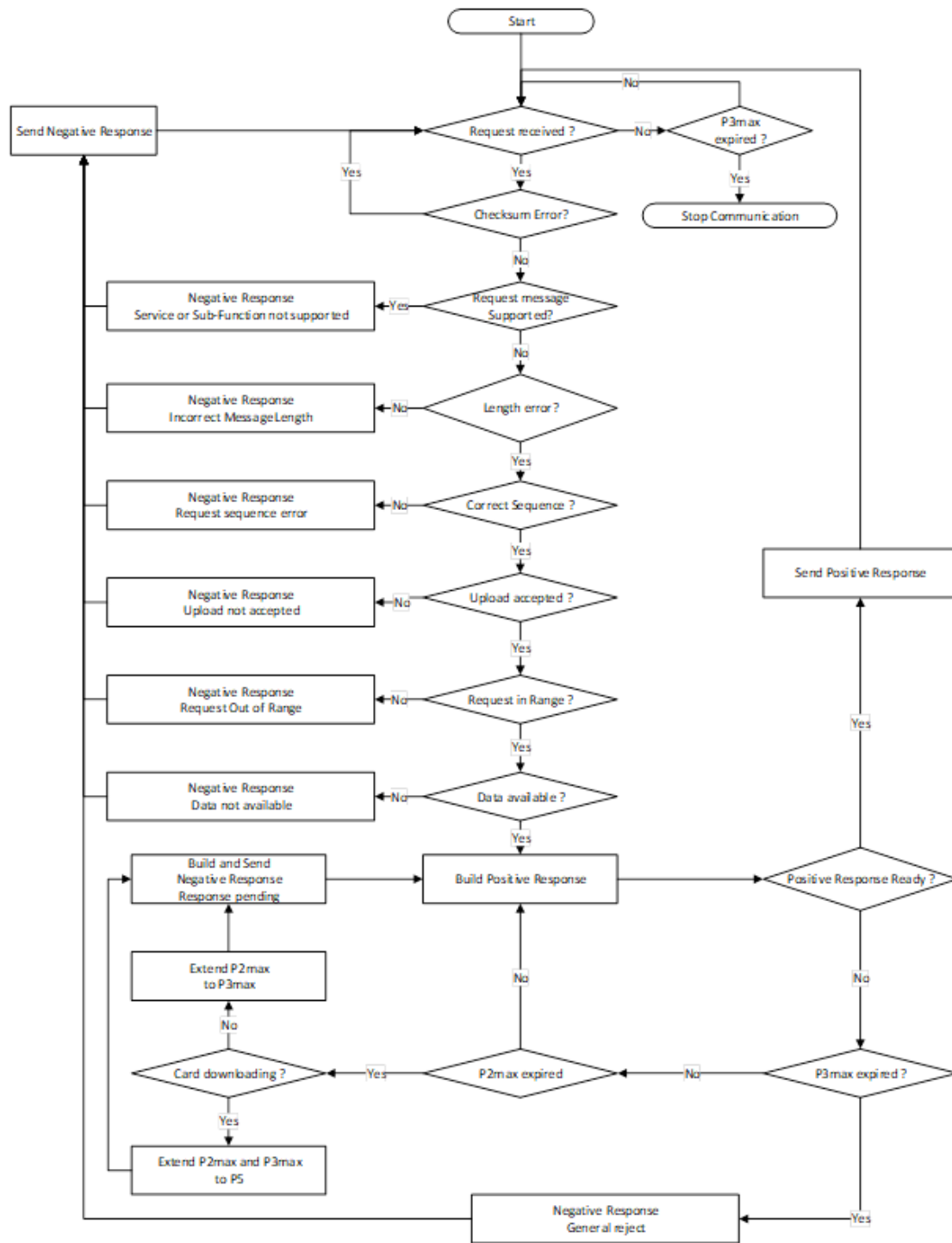


Figure 2 - VU error handling

2. The IDE detects a VU transmission error

DDP_025 For every received message the IDE shall detect timing errors, byte format errors (e.g. start and stop bit violations) and frame errors (wrong number of bytes received, wrong checksum byte).

- DDP_026 The IDE shall detect sequence errors, e.g. incorrect sub message counter increments in successive received messages.
- DDP_027 If the IDE detects an error or there was no response from the VU within a P2max period, the request message will be sent again for a maximum of three transmissions in total. For the purposes of this error detection a sub message acknowledge will be considered as a request to the VU.
- DDP_028 The IDE shall wait at least for a period of P3min before beginning each transmission; the wait period shall be measured from the last calculated occurrence of a stop bit after the error was detected.

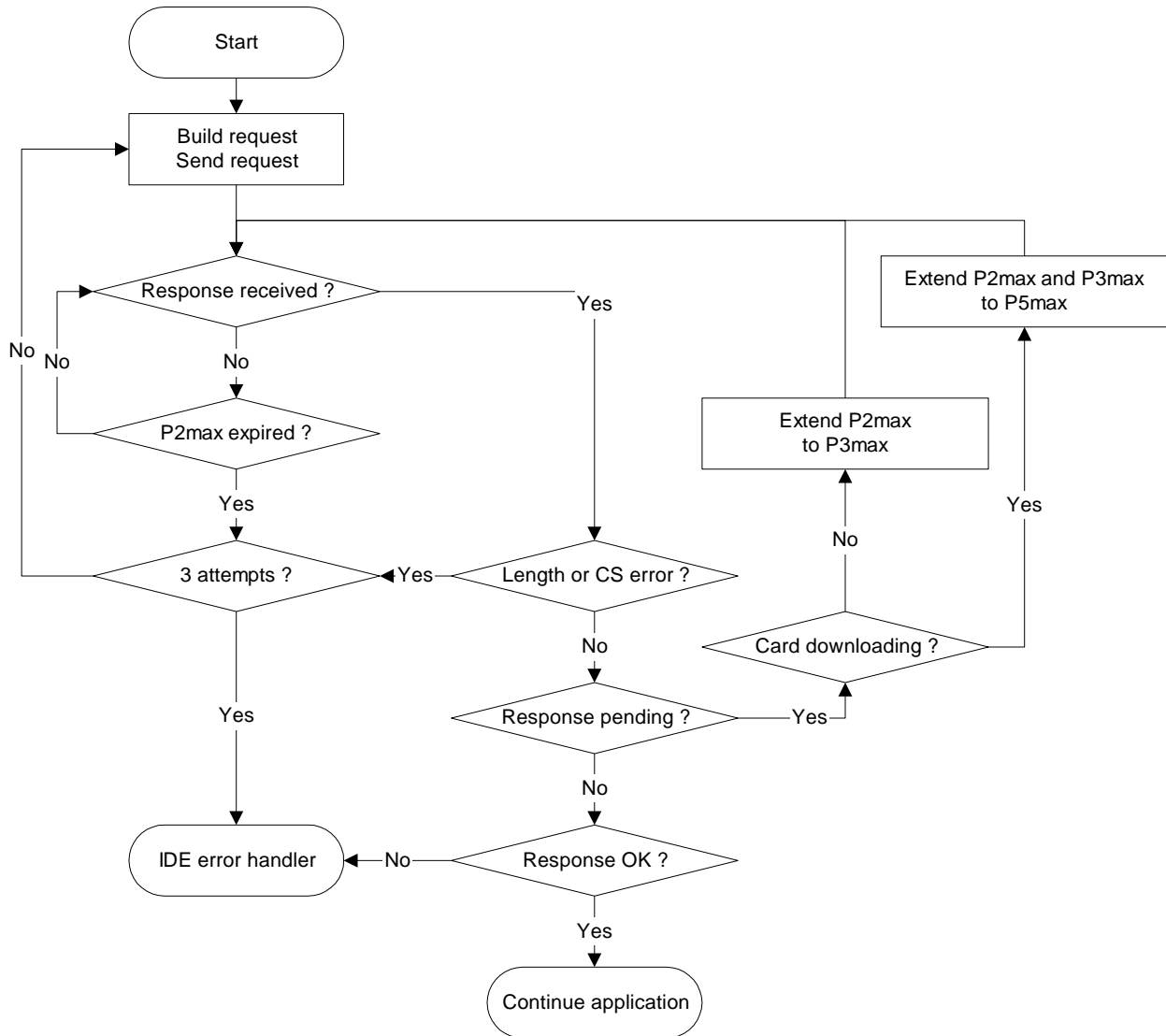


Figure 3 - IDE error handling

2.2.6 Response Message content

This paragraph specifies the content of the data fields of the various positive response messages.

Data elements are defined in ~~Appendix~~ **Sub-appendix 1** data dictionary.

Remark: For generation 2 downloads, each top-level data element is represented by a record array, even if it contains only one record. A record array starts with a header; this header contains the record type, the record size and the number of records. Record arrays are named by "...RecordArray" (with header) in the following tables.

2.2.6.1 Positive Response Transfer Data Download Interface Version

DDP_028a The data field of the ‘Positive Response Transfer Data Download Interface Version’ message shall provide the following data in the following order under the SID 76 Hex, the TREP 00 Hex:

Data structure generation 2, version 2 (TREP 00 Hex)

Data element	Comment
DownloadInterfaceVersion	<p>Generation and version of the VU: 02,02 Hex for Generation 2, version 2.</p> <p>Not supported by Generation 1 and Generation 2, version 1 VU, which shall respond negatively (Sub function not supported, see DDP_018)</p>

2.2.6.2 Positive Response Transfer Data Overview

DDP_029 The data field of the “Positive Response Transfer Data Overview” message shall provide the following data in the following order under the SID 76 Hex, the TREP 01, 21 or 31 Hex and appropriate sub message splitting and counting:

Data structure generation 1 (TREP 01 Hex)

<i>Data element</i>	<i>Comment</i>
MemberStateCertificate	VU Security certificates
VUCertificate	
VehicleIdentificationNumber	Vehicle identification
VehicleRegistrationIdentification	
CurrentDateTime	VU current date and time
VuDownloadablePeriod	Downloadable period
CardSlotsStatus	Type of cards inserted in the VU
VuDownloadActivityData	Previous VU download
VuCompanyLocksData	All company locks stored. If the section is empty, only noOfLocks = 0 is sent.
VuControlActivityData	All control records stored in the VU. If the section is empty, only noOfControls = 0 is sent
Signature	RSA signature of all data (except certificates) starting from VehicleIdentificationNumber down to last byte of last VuControlActivityData..

Data structure generation 2, version 1 (TREP 21 Hex)

Data element	Comment
MemberStateCertificateRecordArray	Member State Contracting Party certificate
VUCertificateRecordArray	VU certificate
VehicleIdentificationNumberRecordArray	Vehicle identification
VehicleRegistrationNumberRecordArray	Vehicle registration number
CurrentDateTimeRecordArray	VU current date and time
VuDownloadablePeriodRecordArray	Downloadable period
CardSlotsStatusRecordArray	Type of cards inserted in the VU
VuDownloadActivityDataRecordArray	Previous VU download
VuCompanyLocksRecordArray	All company locks stored. If the section is empty, an array header with noOfRecords = 0 is sent
VuControlActivityRecordArray	All control records stored in the VU. If the section is empty, an array header with noOfRecords = 0 is sent
SignatureRecordArray	ECC signature of all preceding data except the certificates.

Data structure generation 2, version 2 (TREP 31 Hex)

Data element	Comment
MemberStateCertificateRecordArray	Member State Contracting Party certificate
VUCertificateRecordArray	VU certificate
VehicleIdentificationNumberRecordArray	Vehicle identification
VehicleRegistrationNumberRecordArray	Vehicle registration number
CurrentDateTimeRecordArray	VU current date and time
VuDownloadablePeriodRecordArray	Downloadable period
CardSlotsStatusRecordArray	Type of cards inserted in the VU
VuDownloadActivityDataRecordArray	Previous VU download
VuCompanyLocksRecordArray	All company locks stored. If the section is empty, an array header with noOfRecords = 0 is sent
VuControlActivityRecordArray	All control records stored in the VU. If the section is empty, an array header with noOfRecords = 0 is sent
SignatureRecordArray	ECC signature of all preceding data except the certificates.

2.2.6.23 Positive Response Transfer Data Activities

DDP_030 The data field of the “Positive Response Transfer Data Activities” message shall provide the following data in the following order under the SID 76 Hex, the TREP 02, **22** or **232** Hex and appropriate sub message splitting and counting:

Data structure generation 1 (TREP 02 Hex)

<i>Data element</i>	<i>Comment</i>
TimeReal	Date of day downloaded
OdometerValueMidnight	Odometer at end of downloaded day
VuCardIWData	Cards insertion withdrawal cycles data. <ul style="list-style-type: none"> – If this section contains no available data, only noOfVuCardIWRecords = 0 is sent. – When a VuCardIWRecord lies across 00:00 (card insertion on previous day) or across 24:00 (card withdrawal the following day) it shall appear in full within the two days involved.

<i>Data element</i>	<i>Comment</i>
VuActivityDailyData	Slots status at 00:00 and activity changes recorded for the day downloaded.
VuPlaceDailyWorkPeriodData	Places related data recorded for the day downloaded. If the section is empty, only noOfPlaceRecords = 0 is sent.
VuSpecificConditionData	Specific conditions data recorded for the day downloaded. If the section is empty, only noOfSpecificConditionRecords=0 is sent
Signature	RSA signature of all data starting from TimeReal down to last byte of last specific condition record.

Data structure generation 2, **version 1 (TREP 22 Hex)**

Data element	Comment
DateOfDayDownloadedRecordArray	Date of day downloaded
OdometerValueMidnightRecordArray	Odometer at end of downloaded day
VuCardIWRecordArray	Cards insertion withdrawal cycles data. <ul style="list-style-type: none"> – If this section contains no available data, an array header with noOfRecords = 0 is sent. – When a VuCardIWRecord lies across 00:00 (card insertion on previous day) or across 24:00 (card withdrawal the following day) it shall appear in full within the two days involved.
VuActivityDailyRecordArray	Slots status at 00:00 and activity changes recorded for the day downloaded.
VuPlaceDailyWorkPeriodRecordArray	Places related data recorded for the day downloaded. If the section is empty, an array header with noOfRecords = 0 is sent.
VuGNSSCDRecordArray VuGNSSADRecordArray	GNSS positions of the vehicle if when the continuous accumulated driving time of the driver vehicle reaches a multiple of three hours. If the section is empty, an array header with noOfRecords = 0 is sent.
VuSpecificConditionRecordArray	Specific conditions data recorded for the day downloaded. If the section is empty, an array header with noOfRecords =0 is sent

SignatureRecordArray ECC signature of all preceding data.

Data structure generation 2, version 2 (TREP 32 Hex)

Data element	Comment
DateOfDayDownloadedRecordArray	Date of day downloaded
OdometerValueMidnightRecordArray	Odometer at end of downloaded day
VuCardIWRecordArray	<p>Cards insertion withdrawal cycles data.</p> <ul style="list-style-type: none"> – If this section contains no available data, an array header with noOfRecords = 0 is sent. – When a VuCardIWRecord lies across 00:00 (card insertion on previous day) or across 24:00 (card withdrawal the following day) it shall appear in full within the two days involved.
VuActivityDailyRecordArray	Slots status at 00:00 and activity changes recorded for the day downloaded.
VuPlaceDailyWorkPeriodRecordArray	Places related data recorded for the day downloaded. If the section is empty, an array header with noOfRecords = 0 is sent.
VuGNSSCDRecordArray VuGNSSADRecordArray	GNSS positions of the vehicle when the accumulated driving time of the driver vehicle reaches a multiple of three hours. If the section is empty, an array header with noOfRecords = 0 is sent.
VuSpecificConditionRecordArray	Specific conditions data recorded for the day downloaded. If the section is empty, an array header with noOfRecords = 0 is sent
VuBorderCrossingRecordArray	Border crossings for the day downloaded. If the section is empty, an array header with noOfRecords = 0 is sent.
VuLoadUnloadRecordArray	Load/unload operations for the day downloaded. If the section is empty, an array header with noOfRecords = 0 is sent.
SignatureRecordArray	ECC signature of all preceding data.

2.2.6.43 Positive Response Transfer Data Events and Faults

DDP_031 The data field of the “Positive Response Transfer Data Events and Faults” message shall provide the following data in the following order under the SID 76 Hex, the TREP 03, 23 or 233 Hex and appropriate sub message splitting and counting:

Data structure generation 1 (**TREP 03 Hex**)

Data element	Comment
VuFaultData	All faults stored or on-going in the VU. If the section is empty, only noOfVuFaults = 0 is sent.
VuEventData	All events (except over speeding) stored or on-going in the VU.

<i>Data element</i>	<i>Comment</i>
	If the section is empty, only noOfVuEvents = 0 is sent.
VuOverSpeedingControlData	Data related to last over speeding control (default value if no data).
VuOverSpeedingEventData	All over speeding events stored in the VU. If the section is empty, only noOfVuOverSpeedingEvents = 0 is sent.
VuTimeAdjustmentData	All time adjustment events stored in the VU (outside the frame of a full calibration). If the section is empty, only noOfVuTimeAdjRecords = 0 is sent.
Signature	RSA signature of all data starting from noOfVuFaults down to last byte of last time adjustment record

Data structure generation 2, version 1 (TREP 23 Hex)

Data element	Comment
VuFaultRecordArray	All faults stored or on-going in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.
VuEventRecordArray	All events (except over speeding) stored or on-going in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.
VuOverSpeedingControlDataRecordArray	Data related to last over speeding control (default value if no data).
VuOverSpeedingEventRecordArray	All over speeding events stored in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.
VuTimeAdjustmentRecordArray	All time adjustment events stored in the VU (outside the frame of a full calibration). If the section is empty, an array header with noOfRecords = 0 is sent.
VuTimeAdjustmentGNSSRecordArray	
SignatureRecordArray	ECC signature of all preceding data.

Data structure generation 2, version 2 (TREP 33 Hex)

Data element	Comment
VuFaultRecordArray	All faults stored or on-going in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.
VuEventRecordArray	All events (except over speeding) stored or on-going in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.
VuOverSpeedingControlDataRecordArray	Data related to last over speeding control (default value if no data).
VuOverSpeedingEventRecordArray	All over speeding events stored in the VU. If the section is empty, an array header with noOfRecords = 0 is sent.

VuTimeAdjustmentRecordArray	All time adjustment events stored in the VU (outside the frame of a full calibration). If the section is empty, an array header with noOfRecords = 0 is sent.
SignatureRecordArray	ECC signature of all preceding data.

2.2.6.54 Positive Response Transfer Data Detailed Speed

DDP_032 The data field of the “Positive Response Transfer Data Detailed Speed” message shall provide the following data in the following order under the SID 76 Hex, the TREP 04 or **24** Hex and appropriate sub message splitting and counting:

Data structure generation 1 (TREP 04 Hex)

Data element	Comment
VuDetailedSpeedData	All detailed speed stored in the VU (one speed block per minute during which the vehicle has been moving) 60 speed values per minute (one per second).
Signature	RSA signature of all data starting from noOfSpeedBlocks down to last byte of last speed block.

Data structure generation 2 (TREP 24 Hex)

Data element	Comment
VuDetailedSpeedData VuDetailedSpeed BlockRecordArray	All detailed speed stored in the VU (one speed block per minute during which the vehicle has been moving) 60 speed values per minute (one per second).
SignatureRecordArray	ECC RSA ECC signature of all preceding preceding data starting from noOfSpeedBlocks down to last byte of last speed block.

2.2.6.65 Positive Response Transfer Data Technical Data

DDP_033 The data field of the “Positive Response Transfer Data Technical Data” message shall provide the following data in the following order under the SID 76 Hex, the TREP 05, **25** or ~~235~~ Hex and appropriate sub message splitting and counting:

Data structure generation 1 (TREP 05 Hex)

Data element	Comment
VuIdentification	
SensorPaired	
VuCalibrationData	All calibration records stored in the VU.
Signature	RSA signature of all data starting from vuManufacturerName down to last byte of last VuCalibrationRecord.

Data structure generation 2, version 1 (TREP 25 Hex)

Data element	Comment
VuIdentificationRecordArray	
VuSensorPairedRecordArray	All MS pairings stored in the VU
VuSensorExternalGNSSCoupledRecordArray	All external GNSS facility couplings stored in the VU
VuCalibrationRecordArray	All calibration records stored in the VU.
VuCardRecordArray	All card insertion data stored in the VU.
VuITSConsentRecordArray	
VuPowerSupplyInterruptionRecordArray	
SignatureRecordArray	ECC signature of all preceding data.

Data structure generation 2, version 2 (TREP 35 Hex)

Data element	Comment
VuIdentificationRecordArray	
VuSensorPairedRecordArray	All MS pairings stored in the VU
VuSensorExternalGNSSCoupledRecordArray	All external GNSS facility couplings stored in the VU
VuCalibrationRecordArray	All calibration records stored in the VU.
VuCardRecordArray	All card insertion data stored in the VU.
VuITSConsentRecordArray	
VuPowerSupplyInterruptionRecordArray	
SignatureRecordArray	ECC signature of all preceding data.

2.3. ESM File storage

DDP_034 When a download session has included a VU data transfer, the IDE shall store within one single physical file all data received from the VU during the download session within Positive Response Transfer Data messages. Data stored excludes message headers, sub-message counters, empty sub-messages and checksums but include the SID and TREP (of the first sub-message only if several sub-messages).

3. Tachograph cards downloading protocol

3.1. Scope

This paragraph describes the direct card data downloading of a tachograph card to an IDE. The IDE is not part of the secure environment; therefore no authentication between the card and the IDE is performed.

3.2. Definitions

Download session: Each time a download of the ICC data is performed. The session covers the complete procedure from the reset of the ICC by an IFD until the deactivation of the ICC (withdraw of the card or next reset).

Signed Data File: A file from the ICC. The file is transferred to the IFD in plain text. On the ICC the file is hashed and signed and the signature is transferred to the IFD.

3.3. Card Downloading

DDP_035 The download of a tachograph card includes the following steps:

- Download the common information of the card in the EFs ICC and IC. This information is optional and is not secured with a digital signature.
- ~~(For first and second generation tachograph cards)~~
 - **Download EFs within Tachograph DF:**
 - Download the EFs Card_Certificate ~~(or CardSignCertificate)~~ and CA_Certificate. This information is not secured with a digital signature.
It is mandatory to download these files for each download session.
 - Download the other application data EFs (within Tachograph DF ~~and Tachograph_C2 DF if relevant~~) except EF Card_Download. This information is secured with a digital signature, **using Sub-appendix 11 Common Security Mechanism Part A.**
It is mandatory to download at least the EFs Application_Identification and **Identification for each download session.**
 - **When downloading a driver card it is also mandatory to download the following EFs:**
 - Events_Data,
 - Faults_Data,
 - Driver_Activity_Data,
 - Vehicles_Used,
 - Places,
 - Control_Activity_Data,

Specific_Conditions.

- **(For second generation tachograph cards only)**
 - **Except when a download of a driver card inserted in a VU is performed during drivers' control by a non EU control authority, using a first generation control card ,download EFs within Tachograph_G2 DF:**
 - **Download the EFs CardSignCertificate, CA_Certificate and Link_Certificate (if present). This information is not secured with a digital signature.**
 - It is mandatory to download these files for each download session.
 - **Download the other application data EFs (within Tachograph_G2 DF) except EF Card_Download. This information is secured with a digital signature, using Sub-appendix 11 Common Security Mechanisms Part B.**
 - It is mandatory to download at least the EFs Application_Identification, EF Application_Identification_V2 (if present) and Identification for each download session.**
 - **When downloading a driver card it is also mandatory to download the following EFs:**
 - Events_Data,
 - Faults_Data,
 - Driver_Activity_Data,
 - Vehicles_Used,
 - Places,
 - Control_Activity_Data,**
 - Specific_Conditions,**
 - VehicleUnits_Used,**
 - ~~GNSS_Places (if relevant),~~
 - Places_Authentication, if present,**
 - GNSS_Places_Authentication, if present,**
 - Border_Crossings, if present,**
 - Load_Unload_Operations, if present,**
 - **Load_Type_Entries, if present.**When downloading a driver card, update the LastCardDownload date in EF Card_Download, **in the Tachograph and, if applicable, Tachograph_G2 DFs.**
 - **When downloading a workshop card, reset the calibration counter in EF Card_Download in the Tachograph and, if applicable, Tachograph_G2 DFs.**

- When downloading a workshop card the EF *Sensor_Installation_Data* in the **Tachograph** and, if applicable, **Tachograph_G2** DFs shall not be downloaded.

3.3.1 Initialisation sequence

DDP_036 The IDE shall initiate the sequence as follows:

Card	Direction	IDE / IFD	Meaning / Remarks
	←	Hardware reset	
ATR	⇒		

It is optional to use PPS to switch to a higher baud rate as long as the ICC supports it.

3.3.2 Sequence for un-signed data files

DDP_037 The sequence to download the EFs *ICC*, *IC*, *Card_Certificate* (or *CardSignCertificate*) for **DF Tachograph_G2**, *CA_Certificate* and **Link_Certificate** (for **DF Tachograph_G2** only) is as follows:

Card	Direction	IDE / IFD	Meaning / Remarks
	←	Select File	Select by File identifiers
OK	⇒		
	←	Read Binary	If the file contains more data than the buffer size of the reader or the card the command has to be repeated until the complete file is read.
File Data OK	⇒	Store data to ESM	according to 0 nd with a short EF identifier. 3.4. Data storage format

Note 1: Before selecting the *Card_Certificate* (or *CardSignCertificate*) EF, the *Tachograph* Application must be selected (selection by AID).

Note 2: Selecting and reading a file may also be performed in one step using a *Read Binary* command with a short EF identifier.

3.3.3 Sequence for Signed data files

DDP_038 The following sequence shall be used for each of the following files that has to be downloaded with their signature:

Card	Dir	IDE / IFD	Meaning / Remarks
	←	Select File	
OK	⇒		
	←	Perform Hash of File	Hash Calculates the hash value over the data content of the selected file using the prescribed hash algorithm in accordance with Appendix Sub-appendix 11, part A or B . This command is not an ISO-Command.
	⇒		
	←	Read Binary	If the file contains more data than the buffer of the reader or the card can

Calculate Hash of File and store Hash value temporarily

OK ⇒

← **Read Binary** If the file contains more data than the buffer of the reader or the card can

<i>Card</i>	<i>Dir</i>	<i>IDE / IFD</i>	<i>Meaning / Remarks</i>
			hold, the command has to be repeated until the complete file is read.
File Data OK	⇒	Store received data to ESM	according to 0 nd with a short EF identifier. 3.4. Data storage format
	⇐	PSO: Compute Digital Signature	
Perform Security Operation „Compute Digital Signature“ using the temporarily stored Hash value			
Signature OK	⇒	Append data to the previous stored data on the ESM	according to 0 nd with a short EF identifier. 3.4. Data storage format

Note: Selecting and reading a file may also be performed in one step using a Read Binary command with a short EF identifier. In this case the EF may be selected and read before the command Perform Hash of File is applied.

3.3.4 Sequence for resetting the calibration counter.

DDP_039 The sequence to reset the NoOfCalibrationsSinceDownload counter in the EF Card_Download in a workshop card is the following:

<i>Card</i>	<i>Dir</i>	<i>IDE / IFD</i>	<i>Meaning / Remarks</i>
	⇐	Select File	EF Card_Download Select by File identifiers
OK	⇒		
	⇐	Update Binary	NoOfCalibrationsSinceDownload = '00 00'
resets card download number			
OK	⇒		

Note: Selecting and updating a file may also be performed in one step using an Update Binary command with a short EF identifier.

3.4. Data storage format

3.4.1 Introduction

DDP_040 The downloaded data has to be stored according to the following conditions:

- The data shall be stored transparent. This means that the order of the bytes as well as the order of the bits inside the byte that are transferred from the card has to be preserved during storage.
- All files of the card downloaded within a download session are stored in one file on the ESM.

3.4.2 File format

- DDP_041 The file format is a concatenation of several TLV objects.
- DDP_042 The tag for an EF shall be the FID plus the appendix „00“.
- DDP_043 The tag of an EF's signature shall be the FID of the file plus the appendix „01“.
- DDP_044 The length is a two byte value. The value defines the number of bytes in the value field. The value „FF FF“ in the length field is reserved for future use.
- DDP_045 When a file is not downloaded nothing related to the file shall be stored (no tag and no zero length).
- DDP_046 A signature shall be stored as the next TLV object directly after the TLV object that contains the data of the file.

Definition	Meaning	Length
FID (2 Bytes) „00“	Tag for EF (FID) in the Tachograph DF or for common information of the card	3 Bytes
FID (2 Bytes) „01“	Tag for Signature of EF (FID) in the Tachograph DF	3 Bytes
FID (2 Bytes) „02“	Tag for EF (FID) in the Tachograph_G2 DF	3 Bytes
FID (2 Bytes) „03“	Tag for Signature of EF (FID) in the Tachograph_G2 DF	3 Bytes
xx xx	Length of Value field	2 Bytes

Example of data in a download file on an ESM:

Tag	Length	Value
00 02 00	00 11	Data of EF ICC
C1 00 00	00 C2	Data of EF Card_Certificate
		...
05 05 00	0A 2E	Data of EF Vehicles_Used (in the Tachograph DF)
05 05 01	00 80	Signature of EF Vehicles_Used (in the Tachograph DF)
05 05 02	0A 2E	Data of EF Vehicles_Used in the Tachograph_G2 DF
05 05 03	xx xx	Signature of EF Vehicles_Used in the Tachograph_G2 DF

4. Downloading a tachograph card via a vehicle unit

DDP_047 The VU must allow for downloading the content of a driver card inserted to a connected IDE.

DDP_048 The IDE shall send a “Transfer Data Request Card Download” message to the VU to initiate this mode (see 002.2.2.9).

DDP_049 ~~The~~ First generation driver cards: Data shall be downloaded using the first generation data download protocol, and downloaded data shall have the same format as data downloaded from a first generation vehicle unit.

Second generation driver cards: the VU shall then download the whole card, file by file, in accordance with the card downloading protocol defined in paragraph 0, and forward all data received from the card to the IDE within the

appropriate TLV file format (see -) and encapsulated within a “Positive Response Transfer Data” message.

DDP_050 The IDE shall retrieve card data from the “Positive Response Transfer Data” message (stripping all headers, SIDs, TREPs, sub message counters, and checksums) and store them within one single physical file as described in **paragraph 00.**

DDP_051 The VU shall then, as applicable, update the Control_Activity_Data or the Card_Download file of the driver card

Sub-appendix 8. Calibration Protocol

TABLE OF CONTENT

1. Introduction.....	364
2. Terms, Definitions and References	366
3. Overview of services	367
3.1. Services available	367
3.2. Response codes	368
4. Communication Services	368
4.1. StartCommunication Service	368
4.2. StopCommunication Service	368
4.2.1 Message description	370
4.2.2 Message format	370
4.2.3 Parameter Definition	371
4.3. TesterPresent Service	371
4.3.1 Message description	371
4.3.2 Message format	371
5. Management Services	372
5.1. StartDiagnosticSession service	373
5.1.1 Message description	373
5.1.2 Message format	374
5.1.3 Parameter definition.....	375
5.2. SecurityAccess service	375
5.2.1 Message Description.....	375
5.2.2 Message format - SecurityAccess - requestSeed	376
5.2.3 Message format - SecurityAccess - sendKey	377
6. Data Transmission Services	378
6.1. ReadDataByIdentifier service	378
6.1.1 Message description	378
6.1.2 Message format	379
6.1.3 Parameter Definition	379
6.2. WriteDataByIdentifier service	381
6.2.1 Message description	381
6.2.2 Message format	381
6.2.3 Parameter definition.....	382
7. Control of Test Pulses – Input/Output Control functional unit	382
7.1. InputOutputControlByIdentifier service	382
7.1.1 Message description	383
7.1.2 Message format	383
7.1.3 Parameter definition.....	384
8. routineControlService (Time adjustment)	385
8.1 Message description	385
8.2 Message format.....	387
8. dataRecords formats	387
8.1. Transmitted parameter ranges.....	387
8.2. dataRecords formats	378

1. Introduction

This **sub**-appendix describes how data is exchanged between a vehicle unit and a tester via the K-line which forms part of the calibration interface described in **Appendix Sub-appendix 6**. It also describes control of the input / output signal line on the calibration connector.

Establishing K-line communications is described in Section 4 “**Communication Services**”.

This **sub**-appendix uses the idea of diagnostic “sessions” to determine the scope of K-line control under different conditions. The default session is the “StandardDiagnosticSession” where all data can be read from a vehicle unit but no data can be written to a vehicle unit.-

Selection of the diagnostic session is described in Section Error! Reference source not found. “**Error! Reference source not found.**”.

This **sub**-appendix has to be considered as relevant for both generations of VUs and of workshop cards, in compliance with the interoperability requirements laid down in this Regulation.

CPR_001 The “ECUProgrammingSession” allows data entry into the vehicle unit. In the case of entry of calibration data, the vehicle unit must, in addition be in the CALIBRATION mode of operation.

Data transfer via K-line is described in Section 0 “

6. Data Transmission Services”. Formats of data transferred are detailed in Section 0 “

8. RoutineControl Service (time adjustment)

8.1 Message description

CPR_065a The service RoutineControlService (TimeAdjustment) provides the ability to trigger an alignment of the VU clock to the time provided by the GNSS receiver.

For the service RoutineControlService (TimeAdjustment) execution the VU must be in CALIBRATION mode.

Precondition: it is ensured that the VU is able to receive authenticated position messages from the GNSS receiver.

As long the time adjustment is ongoing, the VU shall respond to the request RoutineControl, subfunction requestRoutineResults, with routineInfo = 0x78.

Note: the time adjustment may take some time. The diagnostic tester shall request the time adjustment status by using the sub-function requestRoutineResults.

8.2 Message format

CPR_065b The message formats for the service RoutineControlService (TimeAdjustment) and its primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Request Sid	31	RC
#6	routineControlType = [startRoutine]	01	RCTP_STR

#7 and #8	routineIdentifier = [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37a - RoutineControl, routine (TimeAdjustment) Request Message subfunction startRoutine

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Positive Response Sid	71	RCPR
#6	routineControlType = [startRoutine]	01	RCTP_STR
#7 and #8	routineIdentifier = [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37b - RoutineControl, routine (TimeAdjustment), subfunction startRoutine, Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Request Sid	31	RC
#6	routineControlType = [requestRoutineResults]	03	RCTP_RRR
#7 and #8	routineIdentifier = [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37c - RoutineControl, routine (TimeAdjustment) Request Message, subfunction requestRoutineResults

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Positive Response Sid	71	RCPR
#6	routineControlType = [requestRoutineResults]	03	RCTP_RRR
#7 and #8	routineIdentifier = [TimeAdjustment]	0100	RI_TA
#9	routineInfo (see Table 37f)	XX	RINF_TA
#10	routineStatusRecord[] = routineStatus#1 (see Table 37g)	XX	RS_TA
#11	Checksum	00-FF	CS

Table 37d - RoutineControl, routine (TimeAdjustment), subfunction requestRoutineResults, Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negativeResponse Service Id	7F	NR
#6	inputOutputControlByIdentifier Request Sid	31	RC

#7	responseCode=[sub-functionNotSupported incorrectMessageLengthOrInvalidFormat conditionsNotCorrect requestOutOfRange]	12 13 22 31	SFNS IMLOIF CNC ROOR
#8	Checksum	00-FF	CS

Table 37e - RoutineControl, routine (TimeAdjustment) Negative Response Message

routineInfo	Hex Value	Description
NormalExitWithResultAvailable	61	The routine was executed completely; additional routine results available.
RoutineExecutionOngoing	78	The requested routine is still executed.

Table 37f - RoutineControl, routine (TimeAdjustment), routineInfo

Hex Value	Test result	Description
01	positive	The time adjustment successfully finished.
02..0F		RFU
10	negative	No GNSS signal reception.
11..7F		RFU
80..FF		Manufacturer specific

Table 37g - RoutineControl, routine (TimeAdjustment), routineStatus

9. dataRecords formats”.
- CPR_002 The “ECUAdjustmentSession” allows the selection of the I/O mode of the calibration I/O signal line via the K-line interface. Control of the calibration I/O signal line is described in section 0 “
7. Control of Test Pulses – Input/Output Control functional unit”.
- CPR_003 Throughout this document the address of the tester is referred to as 't'. Although there may be preferred addresses for testers, the VU shall respond correctly to any tester address. The physical address of the VU is 0xEE.

2. Terms, Definitions and References

The protocols, messages and error codes are principally based on a draft of ISO 14229-1 (Road vehicles – Diagnostic systems – Part 1: Diagnostic services, version 6 of 22 February 2001).

Byte encoding and hexadecimal values are used for the service identifiers, the service requests and responses, and the standard parameters.

The term ‘tester’ refers to the equipment used to enter programming/calibration data into the VU.

The terms ‘client’ and ‘server’ refer to the tester and the VU respectively.

The term ECU means "Electronic Control Unit" and refers to the VU.

References:

ISO 14230-2: Road Vehicles -Diagnostic Systems - Keyword Protocol 2000- Part 2 : Data Link Layer.

First edition: 1999.

~~Vehicles Diagnostic.~~

3. Overview of services

3.1. Services available

The following table provides an overview of the services that will be available in the tachograph and are defined in this document.

CPR_004 The table indicates the services that are available in an enabled diagnostic session.

- The **1st column** lists the services that are available.
- The **2nd column** includes the section number in this **sub**-appendix where of service is further defined.
- The **3rd column** assigns the service identifier values for request messages.
- The **4th column** specifies the services of the "**StandardDiagnosticSession**" (**SD**) which must be implemented in each VU.
- The **5th column** specifies the services of the "**ECUAdjustmentSession**" (**ECUAS**) which must be implemented to allow control of the I/O signal line in the front panel calibration connector of the VU.
- The **6th column** specifies the services of the "**ECUProgrammingSession**" (**ECUPS**) which must be implemented to allow for programming of parameters in the VU.

Diagnostic Service Name	Section No.	Sid Req. Value	Diagnostic Sessions		
			SD	ECUAS	ECUPS
StartCommunication	0	81	■	■	■
StopCommunication	0	82	■		
TesterPresent	0	3E	■	■	■
StartDiagnosticSession	0	10	■	■	■
SecurityAccess	Error! Reference source not found.	27	■	■	■
ReadDataByIdentifier	0	22	■	■	■
WriteDataByIdentifier	0	2E			■
InputOutputControlByIdentifier	0	2F		■	
RoutineControl	8	31		■	■

Table 1 - Service Identifier value summary table

■ This symbol indicates that the service is mandatory in this diagnostic session.

No symbol indicates that this service is not allowed in this diagnostic session.

3.2. Response codes

Response codes are defined for each service.

4. Communication Services

Some services are necessary to establish and maintain communication. They do not appear on the application layer. The services available are detailed in the following table:

Service name	Description
StartCommunication	The client requests to start a communication session with a server(s).
StopCommunication	The client requests to stop the current communication session.
TesterPresent	The client indicates to the server that it is still present.

Table 2 - Communication Services

CPR_005 The StartCommunication Service is used for starting a communication. In order to perform any service, communication must be initialised and the communication parameters need to be appropriate for the desired mode.

4.1. StartCommunication Service

CPR_006 Upon receiving a StartCommunication indication primitive, the VU shall check if the requested communication link can be initialised under the present conditions. Valid conditions for the initialisation of a communication link are described in document ISO 14230-2.

CPR_007 Then the VU shall perform all actions necessary to initialise the communication link and send a StartCommunication response primitive with the Positive Response parameters selected.

CPR_008 If a VU that is already initialised (and has entered any diagnostic session) receives a new StartCommunication Request (e.g. due to error recovery in the tester) the request shall be accepted and the VU shall be reinitialised.

CPR_009 If the communication link cannot be initialised for any reason, the VU shall continue operating as it was immediately prior to the attempt to initialise the communication link.

CPR_010 The StartCommunication Request message must be physically addressed.

CPR_011 **Initialising the VU for services is performed through a 'fast initialisation' method,**

- There is a bus-idle time prior to any activity.
- The tester then sends an initialisation pattern.
- All information which is necessary to establish communication is contained in the response of the VU.

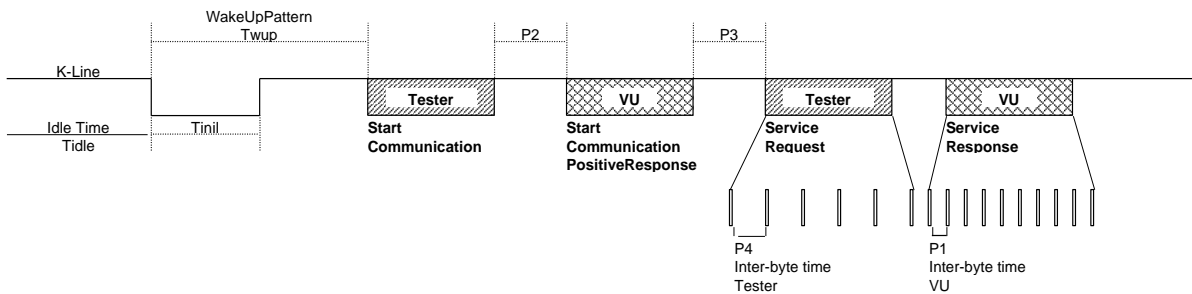
CPR_012 After completion of the initialisation,

- All communication parameters are set to values defined in **Table 4** according to the key bytes.
- The VU is waiting for the first request of the tester.
- The VU is in the default diagnostic mode, i.e. StandardDiagnosticSession.

- The calibration I/O signal line is in the default state, i.e. disabled state.

CPR_014 The data rate on the K-line shall be 10 400 Baud.

CPR_016 The fast initialisation is started by the tester transmitting a Wake up pattern (Wup) on the K-line. The pattern begins after the idle time on K-line with a low time of Tinil. The tester transmits the first bit of the StartCommunication Service after a time of Twup following the first falling edge.



CPR_017 The timing values for the fast initialisation and communications in general are detailed in the tables below. There are different possibilities for the idle time:

- First transmission after power on, Tidle = 300ms.
- After completion of a StopCommunication Service, Tidle = P3 min.
- After stopping communication by time-out P3 max, Tidle = 0.

Parameter	min value	max value
Tinil	25 ± 1 ms	26 ms
Twup	50 ± 1 ms	51 ms

Table 3 - Timing values for fast initialisation

Timing		lower limit values [ms]	upper limit values [ms]
Parameter	Parameter Description	min.	max.
P1	Inter byte time for VU response	0	20
P2	Time between tester request and VU response or two VU responses	25	250
P3	Time between end of VU responses and start of new tester request	55	5000
P4	Inter byte time for tester request	5	20

Table 4 - Communication timing values

CPR_018 The message format for fast initialisation is detailed in the following tables (NOTE: Hex means hexadecimal)

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	81	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	StartCommunication Request Service Id	81	SCR
#5	Checksum	00-FF	CS

Table 5 - StartCommunication Request Message

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	StartCommunication Positive Response Service Id	C1	SCRPR
#6	Key byte 1	EA	KB1
#7	Key byte 2	8F	KB2
#8	Checksum	00-FF	CS

Table 6 - StartCommunication Positive Response Message

CPR_019 There is no negative response to the StartCommunication Request message, if there is no positive response message to be transmitted then the VU is not initialised, nothing is transmitted and it remains in its normal operation.

4.2. StopCommunication Service

4.2.1 Message description

The purpose of this communication layer service is to terminate a communication session.

CPR_020 Upon receiving a StopCommunication indication primitive, the VU shall check if the current conditions allow to terminate this communication. In this case the VU shall perform all actions necessary to terminate this communication.

CPR_021 If it is possible to terminate the communication, the VU shall issue a StopCommunication response primitive with the Positive Response parameters selected, before the communication is terminated.

CPR_022 If the communication cannot be terminated by any reason, the VU shall issue a StopCommunication response primitive with the Negative Response parameter selected.

CPR_023 If time-out of P3max is detected by the VU, the communication shall be terminated without any response primitive being issued.

4.2.2 Message format

CPR_024 The message formats for the StopCommunication primitives are detailed in the following tables.

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	01	LEN

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#5	StopCommunication Request Service Id	82	SPR
#6	Checksum	00-FF	CS

Table 7 - StopCommunication Request Message

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	01	LEN
#5	StopCommunication Positive Response Service Id	C2	SPRPR
#6	Checksum	00-FF	CS

Table 8 - StopCommunication Positive Response Message

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negative Response Service Id	7F	NR
#6	StopCommunication Request Service Identification	82	SPR
#7	responseCode = generalReject	10	RC_GR
#8	Checksum	00-FF	CS

Table 9 - StopCommunication Negative Response Message

4.2.3 Parameter Definition

This service does not require any parameter definition.

4.3. TesterPresent Service

4.3.1 Message description

The TesterPresent service is used by the tester to indicate to the server that it is still present, in order to prevent the server from automatically returning to normal operation and possibly stopping the communication. This service, sent periodically, keeps the diagnostic session / communication active by resetting the P3 timer each time a request for this service is received.

4.3.2 Message format

CPR_079 The message formats for the TesterPresent primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	02	LEN
#5	TesterPresent Request Service Id	3E	TP
#6	Sub Function = responseRequired = [yes no]	01 02	RESPREQ_Y RESPREQ_NO
#7	Checksum	00-FF	CS

Table 10 - TesterPresent Request Message

CPR_80 If the responseRequired parameter is set to 'yes', then the server shall respond with the following positive response message. If set to 'no', then no response is sent by the server.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	01	LEN
#5	TesterPresent Positive Response Service Id	7E	TPPR
#6	Checksum	00-FF	CS

Table 11 - TesterPresent Positive Response Message

CPR_81 The service shall support the following negative responses codes:

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negative Response Service Id	7F	NR
#6	TesterPresent Request Service Identification	3E	TP
#7	responseCode = [SubFunctionNotSupported-InvalidFormat incorrectMessageLength]	12 13	RC_SFNS_IF RC_IML
#8	Checksum	00- FF	CS

Table 12 - TesterPresent Negative Response Message

5. Management Services

The services available are detailed in the following table:

<i>Service name</i>	<i>Description</i>
StartDiagnosticSession	The client requests to start a diagnostic session with a VU.
SecurityAccess	The client requests access to functions restricted to authorised users.

Table 13 - Management Services

5.1. StartDiagnosticSession service

5.1.1 Message description

CPR_025 The service StartDiagnosticSession is used to enable different diagnostic sessions in the server. A diagnostic session enables a specific set of services according to **Table 17**. A session can enable vehicle manufacturer specific services which are not part of this document. Implementation rules shall conform to the following requirements:

- There shall be always exactly one diagnostic session active in the VU,
- The VU shall always start the StandardDiagnosticSession when powered up. If no other diagnostic session is started, then the StandardDiagnosticSession shall be running as long as the VU is powered,
- If a diagnostic session which is already running has been requested by the tester, then the VU shall send a positive response message,
- Whenever the tester requests a new diagnostic session, the VU shall first send a StartDiagnosticSession positive response message before the new session becomes active in the VU. If the VU is not able to start the requested new diagnostic session, then it shall respond with a StartDiagnosticSession negative response message, and the current session shall continue.

CPR_026 A diagnostic session shall only be started if communication has been established between the client and the VU.

CPR_027 The timing parameters defined in **Table 4** shall be active after a successful StartDiagnosticSession with the diagnosticSession parameter set to "StandardDiagnosticSession" in the request message if another diagnostic session was previously active.

5.1.2 Message format

CPR_028 The message formats for the StartDiagnosticSession primitives are detailed in the following tables.

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	02	LEN
#5	StartDiagnosticSession Request Service Id	10	STDS
#6	diagnosticSession = [one value from Table 17]	xx	DS_...
#7	Checksum	00-FF	CS

Table 14 - StartDiagnosticSession Request Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	02	LEN
#5	StartDiagnosticSession Positive Response Service Id	50	STDSPR
#6	diagnosticSession = [same value as in byte #6 Table 14]	xx	DS_...
#7	Checksum	00-FF	CS

Table 15 - StartDiagnosticSession Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	Negative Response Service Id	7F	NR
#6	StartDiagnosticSession Request Service Id	10	STDS
#7	ResponseCode = [subFunctionNotSupported ^a	12	RC_SFNS
	incorrectMessageLength ^b	13	RC_IML
	conditionsNotCorrect ^c	22	RC_CNC
#8	Checksum	00-FF	CS

Table 16 - StartDiagnosticSession Negative Response Message

^a – the value inserted in byte #6 of the request message is not supported, i.e. not in Table 17,

^b – the length of the message is wrong,

^c – the criteria for the request StartDiagnosticSession are not met.

5.1.3 Parameter definition

CPR_029 The parameter *diagnosticSession* (*DS_*) is used by the StartDiagnosticSession service to select the specific behaviour of the server(s). The following diagnostic sessions are specified in this document:

Hex	Description	Mnemonic
81	StandardDiagnosticSession This diagnostic session enables all services specified in Table 1 column 4 "SD" . These services allow reading of data from a server (VU). This diagnostic Session is active after the initialisation has been successfully completed between client (tester) and server (VU). This diagnostic session may be overwritten by other diagnostic sessions specified in this section.	SD

<i>Hex</i>	<i>Description</i>	<i>Mnemonic</i>
85	ECUProgrammingSession This diagnostic session enables all services specified in Table 1 column 6 "ECUPS" . These services support the memory programming of a server (VU) This diagnostic session may be overwritten by other diagnostic sessions specified in this section..	ECUPS
87	ECUAdjustmentSession This diagnostic session enables all services specified in Table 1 column 5 "ECUAS" . These services support the input/output control of a server (VU). This diagnostic session may be overwritten by other diagnostic sessions specified in this section.	ECUAS

Table 17 - Definition of diagnosticSession Values

5.2. SecurityAccess service

Writing of calibration data is not possible unless the VU is in CALIBRATION mode. In addition to insertion of a valid workshop card into the VU, it is necessary to enter the appropriate PIN into the VU before access to the CALIBRATION mode is granted.

When the VU is in CALIBRATION or CONTROL mode, access to the calibration input/output line is also possible.

The SecurityAccess service provides a means to enter the PIN and to indicate to the tester whether or not the VU is in CALIBRATION mode.

It is acceptable that the PIN may be entered through alternative methods.

5.2.1 Message Description

The SecurityAccess service consists of a SecurityAccess "requestSeed" message, eventually followed by a SecurityAccess "sendKey" message. The SecurityAccess service must be carried out after the StartDiagnosticSession service.

- CPR_033 The tester shall use the SecurityAccess "requestSeed" message to check if the vehicle unit is ready to accept a PIN.
- CPR_034 If the vehicle unit is already in CALIBRATION mode, it shall answer the request by sending a "seed" of 0x0000 using the service SecurityAccess Positive Response.
- CPR_035 If the vehicle unit is ready to accept a PIN for verification by a workshop card, it shall answer the request by sending a "seed" greater than 0x0000 using the service SecurityAccess Positive Response.
- CPR_036 If the vehicle unit is not ready to accept a PIN from the tester, either because the workshop card inserted is not valid, or because no workshop card has been inserted, or because the vehicle unit expects the PIN from another method, it shall answer the request with a Negative Response with a response code set to conditionsNotCorrectOrRequestSequenceError.
- CPR_037 The tester shall then, eventually, use the SecurityAccess "sendKey" message to forward a PIN to the Vehicle Unit. To allow time for the card authentication process to take place, the VU shall use the negative response code requestCorrectlyReceived-ResponsePending to extend the time to respond. However, the maximum time to respond shall not exceed 5 minutes. As soon as the requested service has been completed, the VU shall send a positive

response message or negative response message with a response code different from this one. The negative response code requestCorrectlyReceived-ResponsePending may be repeated by the VU until the requested service is completed and the final response message is sent.

- CPR_038 The vehicle unit shall answer to this request using the service SecurityAccess Positive Response only when in CALIBRATION mode.
- CPR_039 In the following cases, the vehicle unit shall answer to this request with a Negative Response with a response code set to:
 - subFunctionNot supported—: Invalid format for the subfunction parameter (accessType),
 - conditionsNotCorrectOrRequestSequenceError: Vehicle unit not ready to accept a PIN entry,
 - invalidKey: PIN not valid and number of PIN checks attempts not exceeded,
 - exceededNumberOfAttempts: PIN not valid and number of PIN checks attempts exceeded,
 - generalReject: Correct PIN but mutual authentication with workshop card failed.

5.2.2 Message format - SecurityAccess – requestSeed

CPR_040 The message formats for the SecurityAccess "requestSeed" primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	02	LEN
#5	SecurityAccess Request Service Id	27	SA
#6	accessType – requestSeed	7D	AT_RSD
#7	Checksum	00-FF	CS

Table 18 – SecurityAccess Request- requestSeed Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	04	LEN
#5	SecurityAccess Positive Response Service Id	67	SAPR
#6	accessType – requestSeed	7D	AT_RSD
#7	Seed High	00-FF	SEEDH
#8	Seed Low	00-FF	SEEDL
#9	Checksum	00-FF	CS

Table 19 – SecurityAccess - requestSeed Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negativeResponse Service Id	7F	NR
#6	SecurityAccess Request Service Id	27	SA
#7	responseCode = [conditionsNotCorrectOrRequestSequenceError incorrectMessageLength]	22 13	RC_CNC RC_IML
#8	Checksum	00-FF	CS

Table 20 – SecurityAccess Negative Response Message

5.2.3 Message format - SecurityAccess – sendKey

CPR_041 The message formats for the SecurityAccess "sendKey" primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	m+2	LEN
#5	SecurityAccess Request Service Id	27	SA
#6	accessType – sendKey	7E	AT_SK
#7 to #m+6	Key #1 (High) ... Key #m (low, m must be a minimum of 4, and a maximum of 8)	xx ... xx	KEY
#m+7	Checksum	00-FF	CS

Table 21 – SecurityAccess Request - sendKey Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	02	LEN
#5	SecurityAccess Positive Response Service Id	67	SAPR
#6	accessType – sendKey	7E	AT_SK
#7	Checksum	00-FF	CS

Table 22 – SecurityAccess - sendKey Positive Response Message

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	NegativeResponse Service Id	7F	NR
#6	SecurityAccess Request Service Id	27	SA
#7	ResponseCode = [generalReject subFunctionNotSupported incorrectMessageLength conditionsNotCorrectOrRequest SequenceError invalidKey exceededNumberOfAttempts requestCorrectlyReceived- ResponsePending]	10 12 13 22 35 36 78	RC_GR RC_SFNS RC_IML RC_CNC RC_IK RC_ENA RC_RCR_RP
#8	Checksum	00-FF	CS

Table 23 – SecurityAccess Negative Response Message

6. Data Transmission Services

The services available are detailed in the following table:

<i>Service name</i>	<i>Description</i>
ReadDataByIdentifier	The client requests the transmission of the current value of a record with access by recordDataIdentifier.
WriteDataByIdentifier	The client requests to write a record accessed by recordDataIdentifier.

Table 24- Data Transmission Services

6.1. ReadDataByIdentifier service

6.1.1 Message description

CPR_050 The ReadDataByIdentifier service is used by the client to request data record values from a server. The data are identified by a recordDataIdentifier. It is the VU manufacturer's responsibility that the server conditions are met when performing this service.

6.1.2 Message format

CPR_051 The message formats for the ReadDataByIdentifier primitives are detailed in the following tables.

<i>Byte #</i>	<i>Parameter Name</i>	<i>Hex Value</i>	<i>Mnemonic</i>
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT

Byte #	Parameter Name	Hex Value	Mnemonic
#3	Source address byte	tt	SRC
#4	Additional length byte	03	LEN
#5	ReadDataByIdentifier Request Service Id	22	RDBI
#6 to #7	recordDataIdentifier = [a value from Table 28]	xxxx	RDI_...
#8	Checksum	00-FF	CS

Table 25 - ReadDataByIdentifier Request Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	m+3	LEN
#5	ReadDataByIdentifier Positive Response Service Id	62	RDBIPR
#6 and #7	recordDataIdentifier = [the same value as bytes #6 and #7 Table 25]	xxxx	RDI_...
#8 to #m+7	dataRecord[] = [data#1 : data#m]	xx : xx	DREC_DATA1 : DREC_DATAm
#m+8	Checksum	00-FF	CS

Table 26 - ReadDataByIdentifier Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	NegativeResponse Service Id	7F	NR
#6	ReadDataByIdentifier Request Service Id	22	RDBI
#7	ResponseCode= [requestOutOfRange incorrectMessageLength conditionsNotCorrect]	31 13 22	RC_ROOR RC_IML RC_CNC
#8	Checksum	00-FF	CS

Table 27 - ReadDataByIdentifier Negative Response Message**6.1.3 Parameter Definition**

CPR_052 The parameter *recordDataIdentifier* (*RDI_*) in the ReadDataByIdentifier request message identifies a data record.

CPR_053 recordDataIdentifier values defined by this document are shown in the table below.

The recordDataIdentifier table consists of ~~four~~**five** columns and multiple lines.

- The **1st column (Hex)** includes the “Hex Value” assigned to the recordDataIdentifier specified in the 3rd column.
- The **2nd column (Data element)** specifies the data element of ~~Appendix~~**Sub-appendix 1** on which the recordDataIdentifier is based (transcoding is sometimes necessary).
- The **3rd column (Description)** specifies the corresponding recordDataIdentifier name.
- The **4th column (Access rights)** specifies the access rights to this recordDataIdentifier.
- The **5th column (Mnemonic)** specifies the mnemonic of this recordDataIdentifier.

<i>Hex</i>	<i>Data element</i>	<i>recordDataIdentifier Name (see format in Section 8.2)</i>	<i>Access (Read/Write)</i>	<i>rights</i>	<i>Mnemonic</i>
F90B	CurrentDateTime	TimeDate	R/W		RDI_TD
F912	HighResOdometer	HighResolutionTotalVehicleDistance	R/W		RDI_HRTVD
F918	K-ConstantOfRecordingEquipment	Kfactor	R/W		RDI_KF
F91C	L-TyreCircumference	LfactorTyreCircumference	R/W		RDI_LF
F91D	W-VehicleCharacteristicConstant	WvehicleCharacteristicFactor	R/W		RDI_WVCF
F921	TyreSize	TyreSize	R/W		RDI_TS
F922	nextCalibrationDate	NextCalibrationDate	R/W		RDI_NCD
F92C	SpeedAuthorised	SpeedAuthorised	R/W		RDI_SA
F97D	vehicleRegistrationNation	RegisteringMemberState	R/W		RDI_RMS
F97E	VehicleRegistrationNumber	VehicleRegistrationNumber	R/W		RDI_VRN
F190	VehicleIdentificationNumber	VIN	R/W		RDI_VIN
F9D0	SensorSerialNumber	MotionSensorSerialNumber	R		RDI_SSN
F9D1	RemoteCommunicationModuleSerialNumber	RemoteCommunicationFacilitySerialNumber	R		RDI_RCSN
F9D2	SensorGNSSSerialNumber	ExternalGNSSFacilitySerialNumber	R		RDI_GSSN
F9D3	SealDataVu	SmartTachographSealsSerialNumber	R/W		RDI_SDV
F9D4	VuSerialNumber	VuSerialNumber	R		RDI_VSN
F9D5	ByDefaultLoadType	ByDefaultLoadType	R/W		RDI_BDLT
F9D6	TachographCardsGen1Suppression	TachographCardsGen1Suppression	R/W		RDI_TCG1S
F9D7	VehiclePosition	VehiclePosition	R		RDI_VP
F9D8	LastCalibrationCountry	CalibrationCountry	R		RDI_CC

Table 28 - Definition of recordDataIdentifier values

CPR_054 The parameter dataRecord (DREC_) is used by the ReadDataByIdentifier positive response message to provide the data record value identified by the recordDataIdentifier to the client (tester). Data formats are specified in section

8. Additional user optional dataRecords including VU specific input, internal and output data may be implemented, but are not defined in this document.

6.2. WriteDataByIdentifier service

6.2.1 Message description

CPR_056 The WriteDataByIdentifier service is used by the client to write data record values to a server. The data are identified by a recordDataIdentifier. It is the VU manufacturer's responsibility that the server conditions are met when performing this service. To update the parameters listed in Table 28 the VU must be in CALIBRATION mode.

6.2.2 Message format

CPR_057 The message formats for the WriteDataByIdentifier primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	m+3	LEN
#5	WriteDataByIdentifier Request Service Id	2E	WDBI
#6 to #7	recordDataIdentifier = [a value from Table 28]	xxxx	RDI_...
#8 to m+7	dataRecord[] = [data#1 : data#m]	xx : xx	DREC_DATA1 : DREC_DATAm
#m+8	Checksum	00-FF	CS

Table 29 - WriteDataByIdentifier Request Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	WriteDataByIdentifier Positive Response Service Id	6E	WDBIPR
#6 to #7	recordDataIdentifier = [the same value as bytes #6 and #7 Table 29]	xxxx	RDI_...
#8	Checksum	00-FF	CS

Table 30 - WriteDataByIdentifier Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC

Byte #	Parameter Name	Hex Value	Mnemonic
#4	Additional length byte	03	LEN
#5	NegativeResponse Service Id	7F	NR
#6	WriteDataByIdentifier Request Service Id	2E	WDBI
#7	ResponseCode= [requestOutOfRange	31	RC_ROOR
	incorrectMessageLength	13	RC_IML
	conditionsNotCorrect]	22	RC_CNC
#8	Checksum	00-FF	CS

Table 31 - WriteDataByIdentifier Negative Response Message

6.2.3 Parameter definition

The parameter *recordDataIdentifier (RDI_)* is defined in **Table 28**.

The parameter *dataRecord (DREC_)* is used by the WriteDataByIdentifier request message to provide the data record values identified by the recordDataIdentifier to the server (VU). Data formats are specified in section 8.

7. Control of Test Pulses – Input/Output Control functional unit

The services available are detailed in the following table:

Service name	Description
InputOutputControlByIdentifier	The client requests the control of an input/output specific to the server.

Table 32 - Input/Output Control functional unit

7.1. InputOutputControlByIdentifier service

7.1.1 Message description

There is a connection via the front connector which allows test pulses to be controlled or monitored using a suitable tester.

CPR_058 This calibration I/O signal line can be configured by K-line command using the InputOutputControlByIdentifier service to select the required input or output function for the line. The available states of the line are:

disabled,

- speedSignalInput, where the calibration I/O signal line is used to input a speed signal (test signal) replacing the motion sensor speed signal, this function is not available in CONTROL mode,
- realTimeSpeedSignalOutputSensor, where the calibration I/O signal line is used to output the speed signal of the motion sensor,
- RTCOutput, where the calibration I/O signal line is used to output the UTC clock signal, this function is not available in CONTROL mode.

- CPR_059 The vehicle unit must have entered an adjustment session and must be in CALIBRATION or CONTROL mode to configure the state of the line. When the VU is in CALIBRATION mode, the four states of the line can be selected (disabled, speedSignalInput, realTimeSpeedSignalOutputSensor, RTCTOutput). When the VU is in CONTROL mode, only two states of the lines can be selected (disabled, realTimeSpeedOutputSensor). On exit of the adjustment session or of the CALIBRATION or CONTROL mode the vehicle unit must ensure the calibration I/O signal line is returned to the 'disabled' (default) state.
- CPR_060 If speed pulses are received at the real time speed signal input line of the VU while the calibration I/O signal line is set to input then the calibration I/O signal line shall be set to output or returned to the disabled state.
- CPR_061 The sequence shall be:
- Establish communications by StartCommunication Service
 - Enter an adjustment session by StartDiagnosticSession Service and be in CALIBRATION or CONTROL mode of operation (the order of these two operation is not important).
 - Change the state of the output by InputOutputControlByIdentifier Service.

7.1.2 Message format

CPR_062 The message formats for the InputOutputControlByIdentifier primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	xx	LEN
#5	InputOutputControlByIdentifier Request Sid	2F	IOCBI
#6 and #7	InputOutputIdentifier = [CalibrationInputOutput]	F960	IOI_CIO
#8 or	ControlOptionRecord = [COR_...
#8 to #9	inputOutputControlParameter - one value from Table 36	xx	IOCP_...
	controlState – one value from Table 37 (see note below)]	xx	CS_...
#9 or #10	Checksum	00-FF	CS

Table 33 - InputOutputControlByIdentifier Request Message

Note: The controlState parameter is present only in some cases (see 7.1.3).

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	xx	LEN

Byte #	Parameter Name	Hex Value	Mnemonic
#5	inputOutputControlByIdentifier Positive Response SId	6F	IOCBIPR
#6 and #7	inputOutputIdentifier = [CalibrationInputOutput]	F960	IOI_CIO
#8 or #8 to #9	controlStatusRecord = [CSR_
	inputOutputControlParameter (same value as byte #8 Table 33)	xx	IOCP_...
	controlState (same value as byte #9 Table 33) (if applicable)	xx	CS_...
#9 or #10	Checksum	00-FF	CS

Table 34 - InputOutputControlByIdentifier Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negativeResponse Service Id	7F	NR
#6	inputOutputControlByIdentifier Request SId	2F	IOCBI
#7	responseCode=[
	incorrectMessageLength	13	RC_IML
	conditionsNotCorrect	22	RC_CNC
	requestOutOfRange	31	RC_ROOR
	deviceControlLimitsExceeded]	7A	RC_DCLE
#8	Checksum	00-FF	CS

Table 35 - InputOutputControlByIdentifier Negative Response Message

7.1.3 Parameter definition

CPR_064 The parameter *inputOutputControlParameter (IOCP_)* is defined in the following table.

Hex	Description	Mnemonic
00	ReturnControlToECU This value shall indicate to the server (VU) that the tester does no longer have control about the calibration I/O signal line.	RCTECU
01	ResetToDefault This value shall indicate to the server (VU) that it is requested to reset the calibration I/O signal line to its default state.	RTD
03	ShortTermAdjustment This value shall indicate to the server (VU) that it is requested to adjust the calibration I/O signal line to the value included in the controlState parameter.	STA

Table 36 - Definition of inputOutputControlParameter values

CPR_065 The parameter *controlState* is present only when the *inputOutputControlParameter* is set to *ShortTermAdjustment* and is defined in the following table:

Mode	Hex Value	Description
Disable	00	I/O line is disabled (default state)
Enable	01	Enable calibration I/O line as <i>speedSignalInput</i>
Enable	02	Enable calibration I/O line as <i>realTimeSpeedSignalOutputSensor</i>
Enable	03	Enable calibration I/O line as <i>RTCOutput</i>

Table 37 - Definition of *controlState* values

8. RoutineControl Service (time adjustment)

8.1 Message description

CPR_065a The service *RoutineControlService (TimeAdjustment)* provides the ability to trigger an alignment of the VU clock to the time provided by the GNSS receiver.

For the service *RoutineControlService (TimeAdjustment)* execution the VU must be in **CALIBRATION** mode.

Precondition: it is ensured that the VU is able to receive authenticated position messages from the GNSS receiver.

As long the time adjustment is ongoing, the VU shall respond to the request *RoutineControl*, subfunction *requestRoutineResults*, with *routineInfo* = 0x78.

Note: the time adjustment may take some time. The diagnostic tester shall request the time adjustment status by using the sub-function *requestRoutineResults*.

8.2 Message format

CPR_065b The message formats for the service *RoutineControlService (TimeAdjustment)* and its primitives are detailed in the following tables.

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Request Sid	31	RC
#6	<i>routineControlType</i> = [startRoutine]	01	RCTP_STR
#7 and #8	<i>routineIdentifier</i> = [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37a - *RoutineControl*, routine (*TimeAdjustment*) Request Message subfunction *startRoutine*

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC

#4	Additional length byte	xx	LEN
#5	RoutineControl Positive Response Sid	71	RCPR
#6	routineControlType = [startRoutine]	01	RCTP_STR
#7 and #8	routineIdentifier= [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37b - RoutineControl, routine (TimeAdjustment), subfunction startRoutine, Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte - physical addressing	80	FMT
#2	Target address byte	EE	TGT
#3	Source address byte	tt	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Request Sid	31	RC
#6	routineControlType = [requestRoutineResults]	03	RCTP_RRR
#7 and #8	routineIdentifier= [TimeAdjustment]	0100	RI_TA
#9	Checksum	00-FF	CS

Table 37c - RoutineControl, routine (TimeAdjustment) Request Message, subfunction requestRoutineResults

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	xx	LEN
#5	RoutineControl Positive Response Sid	71	RCPR
#6	routineControlType = [requestRoutineResults]	03	RCTP_RRR
#7 and #8	routineIdentifier= [TimeAdjustment]	0100	RI_TA
#9	routineInfo (see Table 37f)	XX	RINF_TA
#10	routineStatusRecord[] = routineStatus#1 (see Table 37g)	XX	RS_TA
#11	Checksum	00-FF	CS

Table 37d - RoutineControl, routine (TimeAdjustment), subfunction requestRoutineResults, Positive Response Message

Byte #	Parameter Name	Hex Value	Mnemonic
#1	Format byte – physical addressing	80	FMT
#2	Target address byte	tt	TGT
#3	Source address byte	EE	SRC
#4	Additional length byte	03	LEN
#5	negativeResponse Service Id	7F	NR
#6	inputOutputControlByIdentifier Request Sid	31	RC
#7	responseCode=[sub-functionNotSupported incorrectMessageLengthOrInvalidFormat conditionsNotCorrect requestOutOfRange]	12 13 22 31	SFNS IMLOIF CNC ROOR
#8	Checksum	00-FF	CS

Table 37e - RoutineControl, routine (TimeAdjustment) Negative Response Message

routineInfo	Hex Value	Description
NormalExitWithResultAvailable	61	The routine was executed completely; additional routine results available.
RoutineExecutionOngoing	78	The requested routine is still executed.

Table 37f - RoutineControl, routine (TimeAdjustment), routineInfo

Hex Value	Test result	Description
01	positive	The time adjustment successfully finished.
02..0F		RFU
10	negative	No GNSS signal reception.
11..7F		RFU
80..FF		Manufacturer specific

Table 37g - RoutineControl, routine (TimeAdjustment), routineStatus

9. dataRecords formats

This section details:

- general rules that shall be applied to ranges of parameters transmitted by the vehicle unit to the tester,
- formats that shall be used for data transferred via the Data Transmission Services described in section 6.

CPR_067 All parameters identified shall be supported by the VU.

CPR_068 Data transmitted by the VU to the tester in response to a request message shall be of the measured type (i.e. current value of the requested parameter as measured or observed by the VU).

89.1. Transmitted parameter ranges

CPR_069 **Table 38** defines the ranges used to determine the validity of a transmitted parameter.

CPR_070 The values in the range «error indicator» provide a means for the vehicle unit to immediately indicate that valid parametric data is not currently available due to some type of error in the tachograph.

CPR_071 The values in the range «not available» provide a means for the vehicle unit to transmit a message which contains a parameter that is not available or not supported in that module. The values in the range «not requested» provide a means for a device to transmit a command message and identify those parameters where no response is expected from the receiving device.

CPR_072 If a component failure prevents the transmission of valid data for a parameter, the error indicator as described in **Table 38** should be used in place of that parameter's data. However, if the measured or calculated data has yielded a value that is valid yet exceeds the defined parameter range, the error indicator should not be used. The data should be transmitted using the appropriate minimum or maximum parameter value.

Table 38 – dataRecords ranges

<i>Range Name</i>	<i>1 byte (Hex value)</i>	<i>2 bytes (Hex value)</i>	<i>4 bytes (Hex Value)</i>	<i>ASCII</i>
Valid signal	00 to FA	0000 to FAFF	00000000 to FFFFFFFF	1 to 254
Parameter specific indicator	FB	FB00 to FBFF	FB000000 to FBFFFFFF	none
Reserved range for future indicator bits	FC to FD	FC00 to FDFE	FC000000 to FDEFFFFFFF	none
Error indicator	FE	FE00 to FEFF	FE000000 to FEFFFFFF	0
Not available or not requested	FF	FF00 to FFFF	FF000000 to FFFFFFFF	FF

CPR_073 For parameters coded in ASCII, the ASCII character “*” is reserved as a delimiter.

89.2. dataRecords formats

Table 39 to Table 42 below detail the formats that shall be used via the ReadDataByIdentifier and WriteDataByIdentifier Services.

CPR_074 **Table 39** provides the length, resolution and operating range for each parameter identified by its recordDataIdentifier:

<i>Parameter Name</i>	<i>Data length (bytes)</i>	<i>Resolution</i>	<i>Operating range</i>
TimeDate	8	See details in Table 40	
HighResolutionTotalVehicleDistance	4	5 m/bit gain, 0 m offset	0 to +21 055 406 km
Kfactor	2	0.001 pulse/m /bit gain, 0 offset	0 to 64.255 pulse/m
LfactorTyreCircumference	2	0.125 10 ⁻³ m /bit gain, 0 offset	0 to 8.031 m
WvehicleCharacteristicFactor	2	0.001 pulse/m /bit gain, 0 offset	0 to 64.255 pulse/m
TyreSize	15	ASCII	ASCII
NextCalibrationDate	3	See details in Table 41	
SpeedAuthorised	2	1/256 km/h/bit gain, 0 offset	0 to 250.996 km/h
RegisteringMemberState	3	ASCII	ASCII
VehicleRegistrationNumber	14	See details in Table 42	
VIN	17	ASCII	ASCII
SealDataVu	55	See details in Table 43	
ByDefaultLoadType	1	See details in Table 44	
VuSerialNumber	8	See details in Table 45	
SensorSerialNumber	8	See details in Table 45	
SensorGNSSSerialNumber	8	See details in Table 45	
RemoteCommunicationModuleSerialNumber	8	See details in Table 45	

<i>Parameter Name</i>	<i>Data length (bytes)</i>	<i>Resolution</i>	<i>Operating range</i>
TachographCardsGen1Suppression	2		See details in Table 46
VehiclePosition	14		See details in Table 47
CalibrationCountry	3	ASCII	NationAlpha as defined in Appendix 1

Table 39 – Format of dataRecords

CPR_075 **Table 40** details the formats of the different bytes of the TimeDate parameter :

<i>Byte</i>	<i>Parameter definition</i>	<i>Resolution</i>	<i>Operating range</i>
1	Seconds	0.25 s/bit gain, 0 s offset	0 to 59.75s
2	Minutes	1 min/bit gain, 0 min offset	0 to 59 min
3	Hours	1 h/bit gain, 0 h offset	0 to 23 h
4	Month	1 month/bit gain, 0 month offset	1 to 12 month
5	Day	0.25 day/bit gain, 0 day offset (see NOTE below Table 41)	0.25 to 31.75 day
6	Year	1 year/bit gain, +1985 year offset (see NOTE below Table 41)	1985 to 2235 year
7	Local Minute Offset	1 min/bit gain, -125 min offset	-59 to +59 min
8	Local Hour Offset	1 h/bit gain, -125 h offset	- 23 to +23 h

Table 40 - Detailed format of TimeDate (recordDataIdentifier value # F90B)

CPR_076 **Table 41** details the formats of the different bytes of the NextCalibrationDate parameter.

<i>Byte</i>	<i>Parameter definition</i>	<i>Resolution</i>	<i>Operating range</i>
1	Month	1 month/bit gain, 0 month offset	1 to 12 month
2	Day	0.25 day/bit gain, 0 day offset (see NOTE below)	0.25 to 31.75 day
3	Year	1 year/bit gain, +1985 year offset (see NOTE below)	1985 to 2235 year

Table 41 - Detailed format of NextCalibrationDate (recordDataIdentifier value # F922)

NOTE concerning the use of the “Day” parameter:

- (1) A value of 0 for the date is null. The values 1, 2, 3, and 4 are used to identify the first day of the month; 5, 6, 7, and 8 identify the second day of the month; etc.
- (2) This parameter does not influence or change the hours parameter above.

NOTE concerning the use of the “Year” parameter:

A value of 0 for the year identifies the year 1985; a value of 1 identifies 1986; etc.

CPR_078 **Table 42** details the formats of the different bytes of the VehicleRegistrationNumber parameter:

Byte	Parameter definition	Resolution	Operating range
1	Code Page (as defined in Appendix Sub-appendix 1)	ASCII	01 to applicable
2 – 14	Vehicle Registration Number (as defined in Appendix Sub-appendix 1)	ASCII	VehicleRegistrationNumber

Table 42 - Detailed format of VehicleRegistrationNumber (recordDataIdentifier value # F97E)

CPR_090 Table 43 details the formats of the different bytes of the SealDataVu parameter:

Byte	Parameter definition	Resolution	Operating range
1 – 11	sealRecord1. Format SealRecord as defined in Sub-appendix 1.	not applicable	SealRecord
12 - 22	sealRecord2. Format SealRecord as defined in Sub-appendix 1.	not applicable	SealRecord
23 – 33	sealRecord3. Format SealRecord as defined in Sub-appendix 1.	not applicable	SealRecord
34 – 44	sealRecord4. Format SealRecord as defined in Sub-appendix 1.	not applicable	SealRecord
45 – 55	sealRecord5. Format SealRecord as defined in Sub-appendix 1.	not applicable	SealRecord

Table 43 - Detailed format of SealDataVu (recordDataIdentifier value # F9D3)

NOTE: If there are less than 5 seals available the value of the EquipmentType in all unused sealRecords shall be set to 15, i.e. unused.

CPR_091 Table 44 details the formats of the different bytes of the ByDefaultLoadType parameter:

Byte	Parameter definition	Resolution	Operating range
1	loadType '00'H: Undefined load type '01'H: Goods '02'H: Passengers	not applicable	'00'H to '02'H

Table 44 - Detailed format of ByDefaultLoadType (recordDataIdentifier value # F9D5)

CPR_092 Table 45 details the formats of the different bytes of the VuSerialNumber, SensorSerialNumber, SensorGNSSSerialNumber and RemoteCommunicationModuleSerialNumber parameters:

Byte	Parameter definition	Resolution	Operating range
1	VuSerialNumber, SensorSerialNumber, SensorGNSSSerialNumber and RemoteCommunicationModuleSerialNumber: format ExtendedSerialNumber as defined in Sub-appendix 1.	not applicable	ExtendedSerialNumber

Table 45 - Detailed format of VuSerialNumber, SensorSerialNumber, SensorGNSSSerialNumber and RemoteCommunicationModuleSerialNumber (recordDataIdentifier values # F9D4, F9D0, F9D2, F9D1)

CPR_093 Table 46 details the formats of the different bytes of the TachographCardsGen1Suppression parameter:

Byte	Parameter definition	Resolution	Operating range
1-2	TachographCardsGen1Suppression. Format TachographCardsGen1Suppression as defined in Sub-appendix 1.	not applicable	'0000'H, 'A5E3'H

Table 46 - Detailed format of TachographCardsGen1Suppression (recordDataIdentifier value # F9D6)

CPR_094 Table 47 details the formats of the different bytes of the VehiclePosition parameter.

Byte	Parameter definition	Resolution	Operating range
1 - 4	Time stamp of the vehicle position was determined.	Not applicable	TimeReal
5	GNSS accuracy	Not applicable	GNSSAccuracy
6 - 11	Vehicle position	Not applicable	GeoCoordinates
12	Authentication status	Not applicable	PositionAuthenticationStatus
13	Current country	Not applicable	NationNumeric
14	Current region	Not applicable	RegionNumeric

Table 47 - Detailed format of VehiclePosition (recordDataIdentifier value # F9D7)

Note: after vehicle position update, the update of current country and region may be delayed.

Sub-appendix 9. Type approval

LIST OF MINIMUM REQUIRED TESTS

TABLE OF CONTENT

1.	INTRODUCTION	393
2.	VEHICLE UNIT FUNCTIONAL TESTS	395
3.	MOTION SENSOR FUNCTIONAL TESTS	399
4.	TACHOGRAPH CARDS FUNCTIONAL TESTS.....	402
5.	EXTERNAL GNSS FACILITY TESTS	409
6.	EXTERNAL REMOTE COMMUNICATION FACILITY TESTS.....	411
7.	PAPER FUNCTIONAL TESTS	413
8.	INTEROPERABILITY TESTS	415
9.	OSNMA TESTS	415

1. Introduction

1.1. Type approval

The EC type approval for a ~~recording equipment~~ **control device** (or component) or a tachograph card is based on:

- a **security certification**, based on Common Criteria specifications, against a security target fully compliant with ~~Appendix~~ **Sub-appendix 10** to this ~~Annex (to be completed/modified appendix,~~
- a **functional certification** performed by a ~~Member State Contracting Party~~ authority certifying that the item tested fulfils the requirements of this ~~Annex~~ **appendix** in terms of functions performed, measurement accuracy and environmental characteristics,
- an **interoperability certification** performed by the competent body certifying that the ~~recording equipment~~ **control device** (or tachograph card) is fully interoperable with the necessary tachograph card (or ~~recording equipment~~ **control device**) models (see Chapter 8 of this ~~Annex~~ **appendix**).

This ~~Appendix~~ **sub-appendix** specifies which tests, as a minimum, must be performed by a ~~Member State Contracting Party~~ authority during the functional tests, and which tests, as a minimum, must be performed by the competent body during the interoperability tests. Procedures to follow to carry out the tests or the type of tests are not specified further.

The security certification aspects are not covered by this ~~Appendix~~ **sub-appendix**. If some tests requested for type approval are performed during the security evaluation and certification process, then these tests do not need to be performed again. In this case, only the results of these security tests may be inspected. For information, the requirements expected to be tested (or closely related to tests expected to be performed) during the security certification, are marked with a “*” in this ~~Appendix~~ **sub-appendix**.

The numbered requirements refer to the ~~Appendix~~ **sub-appendix 1** ~~Corpus~~, while the other requirements refer to the other **sub-appendixes** (e.g. PIC_001 refers to requirement PIC_001 of ~~Appendix~~ **sub-appendix 3** Pictograms).

This ~~Annex~~ **sub-appendix** considers separately the type approval of the motion sensor, of the vehicle unit, and of the external GNSS facility as components of the ~~recording equipment~~ **control device**. Each component will get its own type approval certificate in which the other compatible components will be indicated. The functional test of the motion sensor (or external GNSS facility) is done together with the vehicle unit and vice versa.

Interoperability between every model of motion sensor (resp. external GNSS facility) and every model of vehicle unit is not required. In that case the type approval for a motion sensor (resp. external GNSS facility) can be granted only in combination with the type approval of the relevant vehicle unit and vice versa.

The national authority in charge of the functional tests of a vehicle unit or an external GNSS facility must make sure that the embedded GNSS receiver has successfully passed the OSNMA tests specified in this Sub-appendix. These tests are considered to be a part of the functional tests of the vehicle unit or the external GNSS facility.

1.2. References

The following references are used in this ~~Appendix~~ **sub-appendix**:

- IEC 60068-2-1: Environmental testing - Part 2-1: Tests - Test A: Cold
- IEC 60068-2-2: Basic environmental testing procedures; part 2: tests; tests B: dry heat (sinusoidal).
- IEC 60068-2-6: Environmental testing - Part 2: Tests - Test Fc: Vibration
- IEC 60068-2-14: Environmental testing; Part 2-14 : Tests; Test N: Change of temperature
- IEC 60068-2-27: Environmental testing. Part 2: Tests. Test Ea and guidance: Shock
- IEC 60068-2-30: Environmental testing - Part 2-30: Tests - Test Db: Damp heat, cyclic (12 h + 12 h cycle)
- IEC 60068-2-64: Environmental testing - Part 2-64: Tests - Test Fh: Vibration, broadband random and guidance
- IEC 60068-2-78 Environmental testing - Part 2-78: Tests - Test Cab: Damp heat, steady state
- ISO 16750-3 – Mechanical loads (2012-12)
- ISO 16750-4 - Climatic loads(2010-04).
- ISO 20653: Road vehicles – Degree of protection (IP code) – Protection of electrical equipment against foreign objects, water and access
- ISO 10605 :2008 + Technical Corrigendum :2010 + AMD1 :2014 Road vehicles - Test methods for electrical disturbances from electrostatic discharge
- ISO 7637-1 :2002 + AMD1 : 2008 Road vehicles - Electrical disturbances from conduction and coupling - Part 1: Definitions and general considerations.
- ISO 7637-2 Road vehicles - Electrical disturbances from conduction and coupling - Part 2: Electrical transient conduction along supply lines only.
- ISO 7637-3 Road vehicles - Electrical disturbances from conduction and coupling - Part 3: Electrical transient transmission by capacitive and inductive coupling via lines other than supply lines.
- ISO/IEC 7816-1 Identification cards - Integrated circuit(s) cards with contacts - Part 1: Physical characteristics..
- ISO/IEC 7816-2 Information technology - Identification cards - Integrated circuit(s) cards with contacts - Part 2: Dimensions and location of the contacts.
- ISO/IEC 7816-3 Information technology - Identification cards - Integrated circuit(s) cards with contacts - Part 3: Electronic signals and transmission protocol.
- ISO/IEC 10373-1 :2006 + AMD1 :2012 Identification cards - Test methods - Part 1: General characteristics
- ISO/IEC 10373-3 :2010 + Technical Corrigendum :2013 Identification cards - Test methods - Part 3: Integrated circuit cards with contacts and related interface devices
- ISO 16844-3:2004, Cor 1:2006 Road vehicles – Tachograph systems – Part 3: Motion sensor interface (with vehicle units).
- ISO 16844-4 Road vehicles - Tachograph systems - Part 4: CAN interface
- ISO 16844-6 Road vehicles - Tachograph systems - Part 6: Diagnostics
- ISO 16844-7 Road vehicles - Tachograph systems - Part 7: Parameters
- ISO 534 Paper and board -- Determination of thickness, density and specific volume

UN ECE R10 Uniform provisions concerning the approval of vehicles with regard to electromagnetic compatibility (United Nation Economic Commission for Europe)

RGODP JRC Technical Report - Receiver guidelines for OSNMA data processing

2. Vehicle unit functional tests

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
1	Administrative examination		
1.1	Documentation	Correctness of documentation	
1.2	Manufacturer test results	Results of manufacturer test performed during integration. Paper demonstrations.	88, 89, 91
2	Visual inspection		
2.1	Compliance with documentation		
2.2	Identification / markings		224 to 226
2.3	Materials		219 to 223
2.4	Sealing		398, 401 to 405
2.5	External interfaces		
3	Functional tests		
3.1	Functions provided		02 , 03, 04, 05, 07, 382,
3.2	Modes of operation		09 to 11*, 132, 133 134, 135
3.3	Functions and data access rights		12* 13*, 382, 383, 386 to 389
3.4	Monitoring cards insertion and withdrawal		15, 16, 17, 18, 19*, 20*, 132 134
3.5	Speed, position and distance measurement		21 to 37 4
3.6	Time measurement (test performed at 20°C)		38 to 43
3.7	Monitoring driver activities		44 to 53, 132 134
3.8	Monitoring driving status		54, 55, 132 1324
3.9	Driver's Manual entries		56 to 62 c
3.10	Company locks management		63 to 68
3.11	Monitoring control activities		69, 70
3.12	Detection of events and/or faults		71 to 88 a , 132 1324
3.13	Equipment identification data		93*, 94*, 97, 100
3.14	Driver or workshop card insertion and withdrawal data		102* to 104*
3.15	Driver activity data		105* to 107*
3.16	Places and positions data		108* to 112*
3.17	Odometer data		113* to 115*

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
3.18	Detailed speed data		116*
3.19	Events data		117*
3.20	Faults data		118*
3.21	Calibration data		119* to 121*
3.22	Time adjustment data		124*, 125*
3.23	Control activity data		126*, 127*
3.24	Company locks data		128*
3.25	Download activity data		129*
3.26	Specific conditions data		130*, 131*
3.27	Tachograph cards data		132*, 133*
3.28	Border crossings		133a* to 133d*
3.29	Load/unload operation		133e* to 133i*
3.30	Digital map		133j* to 133s*
3.3127	Recording and storing on tachographs cards		134, 135, 136, 137, 138* , 139*, 141*, 142, 143, 144, 145, 146*, 147*, 147a* , 147b* 148*, 149, 150, 150a
3.3228	Displaying		90, 132 134, 149 151 to 166-168, PIC_001, DIS_001
3.3329	Printing		90, 132 134, 167 169 to 179 181, PIC_001, PRT_001 to PRT_014
3.340	Warning		132, 180 134, 182 to 189 191, PIC_001
3.354	Data downloading to external media		90, 132, 190 134, 192 to 194 196
3.362	Remote communication for targeted roadside checks		195 197 to 197 199
3.373	Data exchanges with Output data to additional external devices		198, 199 200, 201
3.384	Calibration		202 to 206*, 383, 384, 386 to 391
3.395	Roadside calibration checking		207 to 209
3.4036	Time adjustment		210 to 212*
3.41	Monitoring border crossings		226a to 226c
3.42	Software update		226d to 226f
3.4337	Non-interference of additional functions		06, 425
3.4438	Motion sensor interface		02, 122

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
3.4539	External GNSS facility		03, 123
3.460	Verify that the VU detects, records and stores the event(s) and/or fault(s) defined by the VU manufacturer when a paired motion sensor reacts to magnetic fields disturbing vehicle motion detection.		217
3.474	Cypher suite and standardized domain parameters		CSM_48, CSM_50
4	Environmental tests		
4.1	Temperature	<p>Verify functionality through: Test according to ISO 16750-4, Chapter 5.1.1.2: Low temperature operation test (72 h @ -20 °C) This test refers to IEC 60068-2-1 : Environmental testing - Part 2-1: Tests - Test A: Cold Test according to ISO 16750-4: Chapter 5.1.2.2: High temperature operation test (72 h at 70 °C) This test refers to IEC 60068-2-2: Basic environmental testing procedures; part 2: tests; tests B: dry heat Test according to ISO 16750-4: Chapter 5.3.2: Rapid change of temperature with specified transition duration (-20°C/70 °C, 20 cycles, dwell time 2h at each temperature) A reduced set of tests (among those defined in section 3 of this table) can be carried out at the lower temperature, the higher temperature and during the temperature cycles</p>	213
4.2	Humidity	<p>Verify that the vehicle unit can withstand a cyclic damp (heat test) through IEC 60068-2-30, test Db, six 24 hours cycles, each temperature varying from +25°C to + 55°C and a relative humidity of 97% at + 25°C and equal to 93% at +55°C</p>	214

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
4.3	Mechanical	<p>1. Sinusoidal vibrations. verify that the vehicle unit can withstand sinusoidal vibrations with the following characteristics: constant displacement between 5 and 11 Hz: 10mm peak constant acceleration between 11 and 300 Hz: 5g This requirement is verified through IEC 60068-2-6, test Fc, with a minimum test duration of 3x12 hours (12 hours per axis) ISO 16750-3 does not require a sinusoidal vibration test for devices located in the decoupled vehicle cab.</p> <p>2. Random vibrations: Test according to ISO 16750-3: Chapter 4.1.2.8: Test VIII: Commercial vehicle, decoupled vehicle cab Random vibration test, 10...2000 Hz, RMS vertical 21.3 m/s², RMS longitudinal 11.8 m/s², RMS lateral 13.1 m/s², 3 axes, 32 h per axis, including temperature cycle - 20...70°C. This test refers to IEC 60068-2-64: Environmental testing - Part 2-64: Tests - Test Fh: Vibration, broadband random and guidance</p> <p>3. Shocks: mechanical shock with 3g half sinus according ISO 16750. The tests described above are performed on different samples of the equipment type being tested.</p>	219
4.4	Protection against water and foreign bodies	Test according to ISO 20653: Road vehicles – Degree of protection (IP code) – Protection of electrical equipment against foreign objects, water and access (No change in parameters); Minimum value IP 40	220, 221
4.5	Over-voltage protection	Verify that the vehicle unit can withstand a power supply of: 24 V versions: 34V at +40°C 1 hour 12V versions: 17V at +40°C 1 hour (ISO 16750-2)	216
4.6	Reverse polarity protection	Verify that the vehicle unit can withstand an inversion of its power supply (ISO 16750-2)	216
4.7	Short-circuit protection	Verify that input output signals are protected against short circuits to power supply and ground (ISO 16750-2)	216
5	EMC tests		

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
5.1	Radiated emissions and susceptibility	Compliance with Regulation ECE R10	218
5.2	Electrostatic discharge	Compliance with ISO 10605:2008 + Technical Corrigendum: 2010 + AMD1: 2014: +/- 4kV for contact and +/- 8kV for air discharge	218
5.3	Conducted transient susceptibility on power supply	For 24V versions: compliance with ISO 7637-2 + ECE Regulation No. 10 Rev. 3: pulse 1a: $V_s = -450V$ $R_i = 50$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +20V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -150V$ $R_i = 50$ ohms pulse 3b: $V_s = +150V$ $R_i = 50$ ohms pulse 4: $V_s = -16V$ $V_a = -12V$ $t_6 = 100ms$ pulse 5: $V_s = +120V$ $R_i = 2,2$ ohms $t_d = 250ms$ For 12V versions: compliance with ISO 7637-1 + ECE Regulation No. 10 Rev. 3: pulse 1: $V_s = -75V$ $R_i = 10$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +10V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -112V$ $R_i = 50$ ohms pulse 3b: $V_s = +75V$ $R_i = 50$ ohms pulse 4: $V_s = -6V$ $V_a = -5V$ $t_6 = 15ms$ pulse 5: $V_s = +65V$ $R_i = 3ohms$ $t_d = 100ms$ Pulse 5 shall be tested only for vehicle units designed to be installed in vehicles for which no external common protection against load dump is implemented For load dump proposal, refer to ISO 16750-2, 4th edition, chapter 4.6.4.	218

3. Motion sensor functional tests

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
1. Administrative examination			
1.1	Documentation	Correctness of documentation	
2. Visual inspection			
2.1.	Compliance with documentation		
2.2.	Identification / markings		225, 226,
2.3	Materials		219 to 223
2.4.	Sealing		398, 401 to 405
3. Functional tests			
3.1	Sensor identification data		95 to 97*
3.2	Motion sensor – vehicle unit pairing		122*, 204

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
3.3	Motion detection Motion measurement accuracy		30 to 35
3.4	Vehicle unit interface		02
3.5	Check that the motion sensor is immune to constant magnetic field. Alternatively, verify that the motion sensor reacts to constant magnetic fields disturbing vehicle motion detection so that a connected VU can detect, record and store sensor faults		217
4. Environmental tests			
4.1	Operating temperature	Verify functionality (as defined in test No 3.3) in temperature range [-40°C; +135°C] through: IEC 60068-2-1 test Ad, with a test duration of 96 hours at the lowest temperature T_{\min} , IEC 60068-2-2 test Bd, with a test duration of 96 hours at the highest temperature T_{\max} Test according to ISO 16750-4: Chapter 5.1.1.2: Low temperature operation test (24 h @ -40 °C) This test refers to IEC 60068-2-1: Environmental testing - Part 2-1: Tests - Test A: Cold IEC 68-2-2 test Bd, with a test duration of 96 hours at the lowest temperature of -40°C. Test according to ISO 16750-4: Chapter 5.1.2.2: High temperature operation test (96 h @ 135 °C) This test refers to IEC 60068-2-2: Basic environmental testing procedures; part 2: tests; tests B: dry heat	213
4.2	Temperature cycles	Test according to ISO 16750-4: Chapter 5.3.2: Rapid change of temperature with specified transition duration (-40°C/135 °C, 20 cycles, dwell time 30 min at each temperature) IEC 60068-2-14: Environmental testing; Part 2-14 : Tests; Test N: Change of temperature	213
4.3	Humidity cycles	Verify functionality (as defined in test No. 3.3) through IEC 60068-2-30, test Db, six 24 hours cycles, each temperature varying from +25°C to + 55°C and a relative humidity of 97% at + 25°C and equal to 93% at +55°C	214

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
4.4	Vibration	ISO 16750-3: Chapter 4.1.2.6: Test VI: Commercial vehicle, engine, gearbox Mixed mode vibration test including a) Sinusoidal vibration test, 20...520 Hz, 11.4 ... 120 m/s ² , ≤ 0.5 oct/min b) Random vibration test, 10...2000 Hz, RMS 177 m/s ² 94 h per axis, including temperature cycle - 20...70°C) This test refers to IEC 60068-2-80: Environmental testing - Part 2-80: Tests - Test Fi: Vibration - Mixed mode	219
4.5	Mechanical shock	ISO 16750-3: Chapter 4.2.3: Test VI: Test for devices in or on the gearbox half-sinusoidal shock, acceleration to be agreed in the range 3000...15000 m/s ² , pulse duration to be agreed, however < 1 ms, number of shocks: to be agreed This test refers to IEC 60068-2-27: Environmental testing. Part 2: Tests. Test Ea and guidance: Shock	219
4.6	Protection against water and foreign bodies	Test according to ISO 20653: Road vehicles – Degree of protection (IP code) – Protection of electrical equipment against foreign objects, water and access (Target value IP 64)	220, 221
4.7	Reverse polarity protection	Verify that the motion sensor can withstand an inversion of its power supply	216
4.8	Short circuit protection	Verify that input output signals are protected against short circuits to power supply and ground	216
5. EMC			
5.1	radiated emissions and susceptibility	Verify compliance with Regulation ECE R10	218
5.2	Electrostatic discharge	Compliance with ISO 10605 :2008 + Technical Corrigendum :2010 + AMD1 :2014: +/- 4kV for contact and +/- 8kV for air discharge	218

No	Test	Description	Related requirements
5.3	Conducted transient susceptibility on data lines)	<p>For 24V versions: compliance with ISO 7637-2 + ECE Regulation No. 10 Rev. 3:</p> <p>pulse 1a: $V_s = -450V$ $R_i = 50$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +20V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -150V$ $R_i = 50$ ohms pulse 3b: $V_s = +150V$ $R_i = 50$ ohms pulse 4: $V_s = -16V$ $V_a = -12V$ $t_6 = 100ms$ pulse 5: $V_s = +120V$ $R_i = 2,2$ ohms $t_d = 250ms$</p> <p>For 12V versions: compliance with ISO 7637-1 + ECE Regulation No. 10 Rev. 3:</p> <p>pulse 1: $V_s = -75V$ $R_i = 10$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +10V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -112V$ $R_i = 50$ ohms pulse 3b: $V_s = +75V$ $R_i = 50$ ohms pulse 4: $V_s = -6V$ $V_a = -5V$ $t_6 = 15ms$ pulse 5: $V_s = +65V$ $R_i = 3$ ohms $t_d = 100ms$</p> <p>Pulse 5 shall be tested only for vehicle units designed to be installed in vehicles for which no external common protection against load dump is implemented</p> <p>For load dump proposal, refer to ISO 16750-2, 4th edition, chapter 4.6.4</p>	218

4. Tachograph cards functional tests

Tests according to this Section 4,

no. 5 'Protocol tests',

no. 6 'Card structure' and

no. 7 'Functional tests'

can be performed by the evaluator or certifier during the Common Criteria (CC) security certification process for the chip module.

Tests number 2.3 and 4.2 are the same. These are the mechanical tests of the combination card body and chip module. If one of these components (card body, chip module) is changed, then these tests are necessary.

No	Test	Description	Related requirements
1.	Administrative examination		
1.1	Documentation	Correctness of documentation	
2	Card Body		
2.1	Printed Design	Make sure that all features for protection and visible data are correctly printed on the card and compliant.	227 to 229, 232, 234 to 236

		<p>[Designator] Annex Appendix 1C, chapter 4.1 'Visible data', 227) The front page shall contain: the words "Driver card" or "Control card" or "Workshop card" or "Company card" printed in capital letters in the official language or languages of the Member State Contracting Party issuing the card, according to the type of the card.</p> <p>[Member State Contracting Party name] Annex Appendix 1C, chapter 4.1 'Visible data', 228) The front page shall contain: the name of the Member State Contracting Party issuing the card (optional).</p> <p>[Sign] Annex Appendix 1C, chapter 4.1 'Visible data', 229) The front page shall contain: the distinguishing sign of the Member State Contracting Party issuing the card, printed in negative in a blue rectangle and encircled by 12 yellow stars.</p> <p>[Enumeration] Annex Appendix 1C, chapter 4.1 'Visible data', 232) The reverse page shall contain: an explanation of the numbered items which appear on the front page of the card.</p> <p>[Colour] Annex Appendix 1C, chapter 4.1 'Visible data', 234) Tachograph cards shall be printed with the following background predominant colours: - driver card: white, - workshop card: red, - control card: blue, - company card: yellow.</p> <p>[Security] Annex Appendix 1C, chapter 4.1 'Visible data', 235) Tachograph cards shall bear at least the following features for protection of the card body against counterfeiting and tampering: - a security design background with fine guilloche patterns and rainbow printing, - at least one two-coloured microprint line.</p> <p>[Markings] Annex Appendix 1C, chapter 4.1 'Visible data', 236) Member States Contracting Parties may add colours or markings, such as national symbols and security features.</p> <p>[Approval mark] Tachograph cards shall contain an approval mark. The approval mark shall be made up of: - a rectangle, within which shall be placed the letter 'e' followed by a distinguishing number or letter for the country which has issued the approval, - an approval number corresponding to the number of the approval certificate for a tachograph card, placed at any point within the immediate proximity of this rectangle.</p>	
--	--	---	--

2.2	Mechanical Tests	<p>[Card size] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [5] Dimension of card, [5.1] Card size, [5.1.1] Card dimensions and tolerances, card type ID-1 Unused card</p> <hr/> <p>[Card edges] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [5] Dimension of card, [5.1] Card size, [5.1.2] Card edges</p> <hr/> <p>[Card construction] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [6] Card construction</p> <hr/> <p>[Card materials] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [7] Card materials</p> <hr/> <p>[Bending stiffness] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.1] Bending stiffness</p> <hr/> <p>[Toxicity] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.3] Toxicity</p> <hr/> <p>[Resistance to chemicals] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.4] Resistance to chemicals</p> <hr/> <p>[Card stability] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.5] Card dimensional stability and warpage with temperature and humidity</p>	240, 243 ISO/IEC 7810
-----	------------------	--	--------------------------

		<p>[Light] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.6] Light</p> <p>[Durability] Annex Appendix 1C, chapter 4.4 'Environmental and electrical specifications', 241) Tachograph cards shall be capable of operating correctly for a five-year period if used within the environmental and electrical specifications.</p> <p>[Peel strength] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.8] Peel strength</p> <p>[Adhesion or blocking] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.9] Adhesion or blocking</p> <p>[Warpage] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.11] Overall card warpage</p> <p>[Resistance to heat] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.12] Resistance to heat</p> <p>[Surface distortions] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.13] Surface distortions</p> <p>[Contamination] Tachograph cards must conform to standard ISO/IEC 7810, Identification cards - Physical characteristics, [8] Card characteristics, [8.14] Contamination and interaction of card components</p>	
--	--	--	--

2.3	Mechanical tests with chip module embedded	<p>[Bending] Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.2] Dynamic bending stress Total number of bending cycles: 4000.</p> <p>[Torsion] Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.3] Dynamic torsional stress Total number of torsion cycles: 4000.</p>	ISO/IEC 7810
3	Module		
3.1	Module	<p>Module is the chip encapsulation and the contact plate.</p> <p>[Surface profile] Tachograph cards must conform to standard ISO/IEC 7816-1:2011, Identification cards - Integrated circuit cards - Part 1: Cards with contacts - Physical characteristics [4.2] Surface profile of contacts</p> <p>[Mechanical strength] Tachograph cards must conform to standard ISO/IEC 7816-1:2011, Identification cards - Integrated circuit cards - Part 1: Cards with contacts - Physical characteristics [4.3] Mechanical strength (of a card and contacts)</p> <p>[Electrical resistance] Tachograph cards must conform to standard ISO/IEC 7816-1:2011, Identification cards - Integrated circuit cards - Part 1: Cards with contacts - Physical characteristics [4.4] Electrical resistance (of contacts)</p> <p>[Dimension] Tachograph cards must conform to standard ISO/IEC 7816-2:2007, Identification cards - Integrated circuit cards - Part 2: Cards with contacts - Dimension and location of the contacts [3] Dimension of the contacts</p> <p>[Location] Tachograph cards must conform to standard ISO/IEC 7816-2:2007, Identification cards - Integrated circuit cards - Part 2: Cards with contacts - Dimension and location of the contacts [4] Number and location of the contacts In case of modules with six contacts, contact 'C4' and 'C8' are not part of this test requirement.</p>	ISO/IEC 7816
4	Chip		
4.1	Chip	<p>[Operating temperature] The Tachograph card chip shall operate in an ambient temperature range between -25 °C and +85 °C.</p>	241 to 244 ECE R10 ISO/IEC 7810 ISO/IEC 10373

		<p>[Temperature and humidity] Annex Appendix 1C, chapter 4.4 'Environmental and electrical specifications', 241) Tachograph cards shall be capable of operating correctly in all the climatic conditions normally encountered in Community territory and at least in the temperature range - 25°C to +70°C with occasional peaks of up to +85°C, "occasional" meaning not more than 4 hours each time and not over 100 times during the life time of the card. The Tachograph cards are exposed in consecutive steps to the following temperatures and humidities for the given time. After each step the Tachograph cards are tested for electrical functionality.</p> <ol style="list-style-type: none"> 1. Temperature of - 20 °C for 2 h. 2. Temperature of +/- 0 °C for 2 h. 3. Temperature of + 20 °C, 50 % RH, for 2 h. 4. Temperature of + 50 °C, 50 % RH, for 2 h. 5. Temperature of + 70 °C, 50 % RH, for 2 h. <p style="padding-left: 20px;">The temperature is increased intermittently to + 85 °C, 50 % RH, for 60 min.</p> <ol style="list-style-type: none"> 6. Temperature of + 70 °C, 85 % RH, for 2 h. <p style="padding-left: 20px;">The temperature is increased intermittently to + 85 °C, 85 % RH, for 30 min.</p>	
		<p>[Humidity] Annex Appendix 1C, chapter 4.4 'Environmental and electrical specifications', 242) Tachograph cards shall be capable of operating correctly in the humidity range 10% to 90%.</p>	
		<p>[Electromagnetic compatibility - EMC] Annex Appendix 1C, chapter 4.4 'Environmental and electrical specifications' 244) During operation, Tachograph cards shall conform to ECE R10 related to electromagnetic compatibility.</p>	
		<p>[Static electricity] Annex Appendix 1C, chapter 4.4 'Environmental and electrical specifications', 244) During operation, Tachograph cards shall be protected against electrostatic discharges. Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.4] Static electricity [9.4.1] Contact IC cards Test voltage: 4000 V.</p>	
		<p>[X-rays] Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.1] X-rays</p>	
		<p>[Ultraviolet light] ISO/IEC 10373-1:2006, Identification cards - Test methods - Part 1: General characteristics [5.11] Ultraviolet light</p>	

		[3-wheel] Tachograph cards must conform to standard ISO/IEC 10373-1:2006/Amd. 1:2012, Identification cards - Test methods - Part 1: General characteristics, Amendment 1 [5.22] ICC - Mechanical strength: 3 wheel test for ICCs with contacts	
		[Wrapping] Tachograph cards must conform to standard MasterCard CQM V2.03:2013 [11.1.3] R-L3-14-8: Wrapping Test Robustness [13.2.1.32] TM-422: Mechanical Reliability: Wrapping Test	
4.2	Mechanical tests chip module embedded in the card body -> same as 2.3	[Bending] Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.2] Dynamic bending stress Total number of bending cycles: 4000. [Torsion] Tachograph cards must conform to standard ISO/IEC 7810:2003/Amd. 1:2009, Identification cards – Physical characteristics, Amendment 1: Criteria for cards containing integrated circuits [9.3] Dynamic torsional stress Total number of torsion cycles: 4000.	ISO/IEC 7810
5	Protocol tests		
5.1	ATR	Check that the ATR is compliant	ISO/IEC 7816-3 TCS_14, TCS_17, TCS_18
5.2	T=0	Check that T=0 protocol is compliant	ISO/IEC 7816-3 TCS_11, TCS_12, TCS_13, TCS_15
5.3	PTS	Check that the PTS command is compliant by setting T=1 from T=0.	ISO/IEC 7816-3 TCS_12, TCS_19, TCS_20, TCS_21
5.4	T=1	Check that T=1 protocol is compliant	ISO/IEC 7816-3 TCS_11, TCS_13, TCS_16
6	Card structure		
6.1		Test that the file structure of the card is compliant by checking the presence of the mandatory files in the card and their access conditions	TCS_22 to TCS_28 TCS_140 to TCS_179
7	Functional tests		
7.1	Normal processing	Test at least once each allowed usage of each command (ex: test the UPDATE BINARY command with CLA = '00', CLA = '0C' and with different P1,P2 and Lc parameters) Check that the operations have actually been performed in the card (ex: by reading the file the command has been performed on)	TCS_29 to TCS_139
7.2	Error messages	Test at least once each error message (as specified in Appendix Sub-appendix 2) for each command Test at least once every generic error (except '6400' integrity errors checked during security certification)	

7.3	Cypher suite and standardized domain parameters		CSM_48, CSM_50
8	Personalisation		
8.1	Optical personalisation	<p>Annex Appendix 1C, chapter 4.1 'Visible data', 230) The front page shall contain: information specific to the card issued.</p> <p>Annex Appendix 1C, chapter 4.1 'Visible data', 231) The front page shall contain: dates using a "dd/mm/yyyy" or "dd.mm.yyyy" format (day, month, year).</p> <p>Annex Appendix 1C, chapter 4.1 'Visible data', 235) Tachograph cards shall bear at least the following features for protection of the card body against counterfeiting and tampering: - in the area of the photograph, the security design background and the photograph shall overlap.</p>	230, 231, 235

5. External GNSS facility tests

No	Test	Description	Related requirements
1.	Administrative examination		
1.1	Documentation	Correctness of documentation	
2.	Visual inspection for external GNSS facility		
2.1.	Compliance with documentation		
2.2.	Identification / markings		224 to 226
2.3	Materials		219 to 223
3.	Functional tests		
3.1	Sensor identification data		98,99
3.2	External GNSS module – vehicle unit coupling		123, 205
3.3	GNSS position S		36, 37
3.4	Vehicle unit interface when the GNSS receiver is external to the Vehicle Unit		03
3.5	Cypher suite and standardized domain parameters		CSM_48, CSM_50
4.	Environmental tests		
4.1	Temperature	<p>Verify functionality through: Test according to ISO 16750-4, Chapter 5.1.1.2: Low temperature operation test (72 h @ -20 °C) This test refers to IEC 60068-2-1: Environmental testing - Part 2-1: Tests - Test A: Cold</p> <p>Test according to ISO 16750-4: Chapter 5.1.2.2: High temperature operation test (72 h @ 70 °C) This test refers to IEC 60068-2-2: Basic environmental testing procedures; part 2: tests; tests B: dry heat</p> <p>Test according to ISO 16750-4: Chapter 5.3.2: Rapid change of temperature with specified transition duration (-20°C/70 °C, 20 cycles, dwell time 1 h at each temperature) A reduced set of tests (among those defined in section 3 of this table) can be carried out at the lower temperature, the higher temperature and during the temperature cycles</p>	213

4.2	Humidity	Verify that the vehicle unit can withstand a cyclic damp (heat test) through IEC 60068-2-30, test Db, six 24 hours cycles, each temperature varying from +25°C to +55°C and a relative humidity of 97% at +25°C and equal to 93% at +55°C	214
4.3	Mechanical	<p>Sinusoidal vibrations. verify that the vehicle unit can withstand sinusoidal vibrations with the following characteristics: constant displacement between 5 and 11 Hz: 10mm peak constant acceleration between 11 and 300 Hz: 5g This requirement is verified through IEC 60068-2-6, test Fc, with a minimum test duration of 3x12 hours (12 hours per axis) ISO 16750-3 does not require a sinusoidal vibration test for devices located in the decoupled vehicle cab. Random vibrations: Test according to ISO 16750-3: Chapter 4.1.2.8: Test VIII: Commercial vehicle, decoupled vehicle cab Random vibration test, 10...2000 Hz, RMS vertical 21.3 m/s², RMS longitudinal 11.8 m/s², RMS lateral 13.1 m/s², 3 axes, 32 h per axis, including temperature cycle -20...70°C. This test refers to IEC 60068-2-64: Environmental testing - Part 2-64: Tests - Test Fh: Vibration, broadband random and guidance Shocks: mechanical shock with 3g half sinus according ISO 16750.</p> <p>The tests described above are performed on different samples of the equipment type being tested.</p>	219
4.4	Protection against water and foreign bodies	Test according to ISO 20653: Road vehicles – Degree of protection (IP code) – Protection of electrical equipment against foreign objects, water and access (No change in parameters)	220, 221
4.5	Over-voltage protection	Verify that the vehicle unit can withstand a power supply of: 24 V versions: 34V at +40°C 1 hour 12V versions: 17V at +40°C 1 hour (ISO 16750-2, chapter 4.3)	216
4.6	Reverse polarity protection	Verify that the vehicle unit can withstand an inversion of its power supply (ISO 16750-2, chapter 4.7)	216
4.7	Short-circuit protection	Verify that input output signals are protected against short circuits to power supply and ground (ISO 16750-2, chapter 4.10)	216
5	EMC tests		
5.1	Radiated emissions and susceptibility	Compliance with Regulation ECE R10	218
5.2	Electrostatic discharge	Compliance with ISO 10605 :2008 + Technical Corrigendum :2010 + AMD1 :2014: +/- 4kV for contact and +/- 8kV for air discharge	218

5.3	Conducted transient susceptibility on power supply	<p>For 24V versions: compliance with ISO 7637-2 + ECE Regulation No. 10 Rev. 3: pulse 1a: $V_s = -450V$ $R_i = 50$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +20V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -150V$ $R_i = 50$ ohms pulse 3b: $V_s = +150V$ $R_i = 50$ ohms pulse 4: $V_s = -16V$ $V_a = -12V$ $t_6 = 100ms$ pulse 5: $V_s = +120V$ $R_i = 2,2$ ohms $t_d = 250ms$</p> <p>For 12V versions: compliance with ISO 7637-1 + ECE Regulation No. 10 Rev. 3: pulse 1: $V_s = -75V$ $R_i = 10$ ohms pulse 2a: $V_s = +37V$ $R_i = 2$ ohms pulse 2b: $V_s = +10V$ $R_i = 0,05$ ohms pulse 3a: $V_s = -112V$ $R_i = 50$ ohms pulse 3b: $V_s = +75V$ $R_i = 50$ ohms pulse 4: $V_s = -6V$ $V_a = -5V$ $t_6 = 15ms$ pulse 5: $V_s = +65V$ $R_i = 3$ohms $t_d = 100ms$</p> <p>Pulse 5 shall be tested only for vehicle units designed to be installed in vehicles for which no external common protection against load dump is implemented</p> <p>For load dump proposal, refer to ISO 16750-2, 4th edition, chapter 4.6.4.</p>	218
-----	--	--	-----

6. External remote communication facility tests

No	Test	Description	Related requirements
1.	Administrative examination		
1.1	Documentation	Correctness of documentation	
2.	Visual inspection		
2.1.	Compliance with documentation		
2.2.	Identification / markings		225, 226
2.3	Materials		219 to 223
3.	Functional tests		
3.1	Remote communication for targeted roadside checks		4, 197 to 199
3.2	Recording and storing in data memory		91
3.3	Communication with Vehicle Unit		Sub-appendix 14, DCS_66 to DCS_70, DCS_71 to DCS_76
4.	Environmental tests		

4.1	Temperature	<p>Verify functionality through:</p> <p>Test according to ISO 16750-4, Chapter 5.1.1.2: Low temperature operation test (72 h @ -20 °C) This test refers to IEC 60068-2-1: Environmental testing - Part 2-1: Tests - Test A: Cold</p> <p>Test according to ISO 16750-4: Chapter 5.1.2.2: High temperature operation test (72 h @ 70 °C) This test refers to IEC 60068-2-2: Basic environmental testing procedures; part 2: tests; tests B: dry heat</p> <p>Test according to ISO 16750-4: Chapter 5.3.2: Rapid change of temperature with specified transition duration (-20°C/70 °C, 20 cycles, dwell time 1 h (↔) at each temperature) A reduced set of tests (among those defined in section 3 of this table) can be carried out at the lower temperature, the higher temperature and during the temperature cycles</p>	213
4.2	Protection against water and foreign bodies	Test according to ISO 20653: Road vehicles – Degree of protection (IP code) – Protection of electrical equipment against foreign objects, water and access (targeted value IP40)	220, 221
5	EMC tests		
5.1	Radiated emissions and susceptibility	Compliance with Regulation ECE R10	218
5.2	Electrostatic discharge	Compliance with ISO 10605 :2008 + Technical Corrigendum :2010 + AMD1 :2014: +/- 4kV for contact and +/- 8kV for air discharge	218

5.3	Conducted transient susceptibility on power supply	<p>For 24V versions: compliance with ISO 7637-2 + ECE Regulation No. 10 Rev. 3: pulse 1a: $V_s=-450V$ $R_i=50$ ohms pulse 2a: $V_s=+37V$ $R_i=2$ ohms pulse 2b: $V_s=+20V$ $R_i=0,05$ ohms pulse 3a: $V_s=-150V$ $R_i=50$ ohms pulse 3b: $V_s=+150V$ $R_i=50$ ohms pulse 4: $V_s=-16V$ $V_a=-12V$ $t_6=100ms$ pulse 5: $V_s=+120V$ $R_i=2,2$ ohms $t_d=250ms$</p> <p>For 12V versions: compliance with ISO 7637-1 + ECE Regulation No. 10 Rev. 3: pulse 1: $V_s=-75V$ $R_i=10$ ohms pulse 2a: $V_s=+37V$ $R_i=2$ ohms pulse 2b: $V_s=+10V$ $R_i=0,05$ ohms pulse 3a: $V_s=-112V$ $R_i=50$ ohms pulse 3b: $V_s=+75V$ $R_i=50$ ohms pulse 4: $V_s=-6V$ $V_a=-5V$ $t_6=15ms$ pulse 5: $V_s=+65V$ $R_i=3ohms$ $t_d=100ms$</p> <p>Pulse 5 shall be tested only for vehicle units designed to be installed in vehicles for which no external common protection against load dump is implemented</p> <p>For load dump proposal, refer to ISO 16750-2, 4th edition, chapter 4.6.4.</p>	218
-----	--	--	-----

7. Paper functional tests

No	Test	Description	Related requirements
1.	Administrative examination		
1.1	Documentation	Correctness of documentation	
2	General Tests		
2.1	Number of characters per line	Visual inspection of printouts.	172
2.2	Minimum character size	Visual inspection of printout and character inspection.	173
2.3	Supported character sets	The printer shall support characters specified in Appendix Sub-appendix 1 Chapter 4 "Character sets".	174
2.4	Printouts definition	Check of tachograph type approval and visual inspection of printouts	174
2.5	Legibility and identification of printouts	Inspection of printouts Demonstrated by test reports and test protocols by manufacturer. All homologation number(s) of tachographs with which the printer paper may be used are imprinted on the paper.	175, 177, 178
2.6	Addition of handwritten notes	Visual inspection: Field for signature of the driver is available. Fields for additional other handwritten entries are available.	180
2.7	Additional details on paper faces.	Paper's face and reverse side may feature additional details and information. These additional details and information may not interfere with the legibility of the printouts. Visual inspection.	177, 178

3 Storage Tests			
3.1	Dry Heat	Preconditioning: 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 72 hours at +70 °C ± 2°C Recovery: 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178 IEC 60068-2-2-Bb
2.2	Damp Heat	Preconditioning: 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 144 hours at +55°C ± 2°C and 93% ± 3% r.h. Recovery: 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178 IEC 60068-2-78-Cab
4 Paper In-Service Tests			
4.1	Humidity resistance background (unprinted paper)	Preconditioning : 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 144 hours at +55°C ± 2°C and 93% ± 3% r.h. Recovery : 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178 IEC 60068-2-78-Cab
4.2	Printability	Preconditioning : 24 hours at +40°C ± 2°C / 93% ±3% relative humidity Test environment: printout produced at +23°C ± 2°C Recovery : 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178
4.3	Heat resistance	Preconditioning : 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 2 hours at +70°C ± 2°C, dry heat Recovery : 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178 IEC 60068-2-2-Bb
4.4	Low temperature resistance	Preconditioning : 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 24 hours -20 °C ± 3°C, dry cold Recovery : 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178 ISO 60068-2-1-Ab
4.5	Light resistance	Preconditioning : 16 hours at +23°C ± 2°C / 55% ±3% relative humidity Test environment: 100 hours under 5000 Lux illumination at +23°C ± 2°C / 55% ± 3% relative humidity Recovery : 16 hours at +23°C ± 2°C / 55% ± 3% relative humidity	176, 178

Legibility criteria for tests 3.x and 4.x :

Printout legibility is assured if optical densities comply with the following limits:

Printed characters: min. 1,0

Background (unprinted paper): max. 0,2

Optical densities of the resulting printouts shall be measured according to DIN EN ISO 534.

Printouts shall show no dimensional changes and remain clearly legible.

8. Interoperability tests

No	Test	Description
8.1 Interoperability tests between vehicle units and tachograph cards		
1	Mutual authentication	Check that the mutual authentication between the vehicle unit and the tachograph card runs normally
2	Write/read tests	Execute a typical activity scenario on the vehicle unit. The scenario shall be adapted to the type of card being tested and involve writings in as many EFs as possible in the card Verify through a vehicle unit downloading that all corresponding recordings have been properly made Verify through a card downloading that all corresponding recordings have been properly made Verify through daily printouts that all corresponding recordings can be properly read
8.2 Interoperability tests between vehicle units and motion sensors		
1	Pairing	Check that the pairing between the vehicle units and the motion sensors runs normally
2	Activity tests	Execute a typical activity scenario on the motion sensor. The scenario shall involve a normal activity and creating as many events or faults as possible. Verify through a vehicle unit downloading that all corresponding recordings have been properly made Verify through a card downloading that all corresponding recordings have been properly made Verify through a daily printout that all corresponding recordings can be properly read
8.3 Interoperability tests between vehicle units and external GNSS facilities (when applicable)		
1	Mutual Authentication	Check that the mutual authentication (coupling) between the vehicle unit and the external GNSS module runs normally.
2	Activity tests	Execute a typical activity scenario on the external GNSS. The scenario shall involve a normal activity and creating as many events or faults as possible. Verify through a vehicle unit downloading that all corresponding recordings have been properly made Verify through a card downloading that all corresponding recordings have been properly made Verify through a daily printout that all corresponding recordings can be properly read

9. OSNMA tests

9.1. Introduction

This chapter describes the tests to prove the correct implementation of OSNMA in the GNSS receiver. Since satellite signal authentication is carried out solely by the GNSS receiver with independence of any other component of the tachograph, the tests set out in this chapter may be performed on the GNSS receiver as a stand-alone element. In this case, the tachograph manufacturer shall present a report to the type-approval authorities providing details about the development and results of the tests that are performed under the responsibility of the GNSS receiver manufacturer.

9.2 Applicable conditions

- The pass/fail criteria defined in the OSNMA tests shall be considered valid only for the identified testing conditions.
- The criteria might be revised at the moment of the OSNMA service declaration and considering the associated service performance commitments.

9.3. Definitions and acronyms

9.3.1 Definitions

GNSS cold/warm/hot start: refers to the start condition of a GNSS receiver based on the availability of time (T), current almanac (A) and ephemeris (E), position (P):

- GNSS Cold Start: none
- GNSS Warm Start: T, A, P
- GNSS Hot Start: T, A, E, P

OSNMA cold/warm/hot start: refers to the start condition of the OSNMA function based on the availability of the Public Key (P) and DSM-KROOT (K) information (as defined in the OSNMA Receiver Guidelines referred to in Sub-appendix 12):

- OSNMA Cold Start: none
- OSNMA Warm Start: P
- OSNMA Hot Start: P, K

9.3.2 Acronyms

ADKD	Authentication Data & Key Delay
DSM-KROOT	Digital Signature Message KROOT
GNSS	Global Navigation Satellite System
KROOT	Root Key of the TESLA key chain
MAC	Message Authentication Code
NMAC	Number of MAC & key blocks (per 30 seconds)
OSNMA	Galileo-Open Service Navigation Message Authentication
SLMAC	Slow MAC
TESLA	Timed Efficient Stream Loss-tolerant Authentication (Protocol used in OSNMA)

9.4. Equipment for the generation of the GNSS signals

The generation of the GNSS signals necessary for the performance of the GNSS can be carried out using a multi-constellation GNSS signal generator and user-defined navigation messages, scenario start time and duration. Alternatively, a radiofrequency signal re-player capable of playing back GNSS signal samples from files can be used. Typical bit depth and sampling rate are respectively 4 bits I/Q and 10MHz.

It is assumed that the GNSS receiver has interfaces to command the clearing of the receiver memory (to independently erase the public key, KROOT, clock information, position information, ephemeris and almanac), to set the receiver local time realisation for the OSNMA timing verification requirement, and to load the cryptographic information. These commands may be limited to test purposes and therefore may not be available for the receiver nominal operation.

9.5 Test conditions

9.5.1 GNSS conditions

The simulated or replayed GNSS signals will have the following features:

- Static user receiver scenario;
- E1/L1 frequency;
- At least 4 satellites transmitting OSNMA, with elevation angle greater than 5°;
- Duration as required for each test;
- Constant navigation ephemerides from the satellites during the test.

9.5.2 OSNMA conditions

The OSNMA message transmitted in the RF signal will have the following features:

- An HKROOT message with OSNMA Status set to Operational or Test and a fixed DSM-KROOT of 8 blocks for the chain in force;
- At least 4 satellites transmitting OSNMA;
- A MACK message with one MACK block (i.e. NMACK=1), and at least one ADKD=0 and one ADKD=12 per satellite and MACK block;
- A tag size of 40 bits;
- The minimum equivalent tag length as required in the OSNMA Receiver Guidelines (currently 80 bits).

Except when noted, the internal receiver time realisation shall be known with sufficient accuracy and properly aligned with the simulated time. This guarantees that the OSNMA initial time synchronisation requirement is fulfilled for each test condition, i.e., nominal synchronization for all but the SLMAC test. See the OSNMA Receiver Guidelines for more details on the time initialization.

9.6. Tests specification

No	Test	Description	Related requirements
1.	Administrative examination		
1.1	Documentation	Correctness of documentation	
2	General Tests		
2.1	OSNMA hot start	<p>Objective: verify that the GNSS receiver computes a position with OSNMA after a hot start.</p> <p>Procedure: the GNSS receiver starts in GNSS and OSNMA hot start conditions and acquires the signals of visible satellites.</p> <p>The receiver authenticates the navigation data with OSNMA (ADKD = 0) and provides a position with authenticated data.</p> <p>Pass/fail criteria: the receiver computes an authenticated position fix within 160 seconds.</p>	Appendix 12, GNS_3b

2.2	OSNMA warm start	<p>Objective: verify that the GNSS receiver computes a position with OSNMA after a warm start.</p> <p>Procedure: before starting the test, the ephemeris and KROOT information shall be erased from the GNSS receiver memory in order to force a warm GNSS and OSNMA start.</p> <p>The GNSS receiver starts and acquires the signals of the visible satellites.</p> <p>The DSM-KROOT is received and verified.</p> <p>The receiver authenticates the navigation data with OSNMA (ADKD=0) and provides a position with authenticated data.</p> <p>Pass/fail criteria: the receiver computes an authenticated valid position fix within 430 seconds.</p>	Appendix 12, GNS_3b
2.3	OSNMA warm start with SLMAC	<p>Objective: verify that the GNSS receiver computes a position with OSNMA after a warm start with a time initialisation requiring SLMAC mode, as defined in the OSNMA Receiver Guidelines.</p> <p>Procedure: the internal receiver time realisation shall be configured in order to have an initial time uncertainty of a value between 2 and 2.5 minutes so that, according to OSNMA Receiver Guidelines, the Slow MAC mode is activated.</p> <p>Before starting the tests, the ephemeris and KROOT information shall be erased from the GNSS receiver memory in order to force a warm GNSS and OSNMA start.</p> <p>The GNSS receiver starts and acquires the signals of the visible satellites.</p> <p>The DSM-KROOT is received and verified.</p> <p>The receiver authenticates the navigation data with only OSNMA Slow MAC (ADKD=12) and provides a position with authenticated data.</p> <p>Pass/fail criteria: the receiver computes an authenticated valid position fix within 730 seconds.</p>	Appendix 12, GNS_3b
2.4	OSNMA hot start with replayed signal	<p>Objective: verify that the GNSS receiver detects a replayed signal.</p> <p>Procedure: the GNSS receiver starts in GNSS and OSNMA hot start conditions and acquires the signals of visible satellites.</p> <p>The receiver authenticates the navigation data with OSNMA (ADKD=0) and provides a position with authenticated data.</p> <p>Once the receiver provides PVT solution with authenticated data, it is switched off.</p> <p>A replayed signal with a delay of 40 seconds with respect to the previous one is simulated, and the receiver is switched on.</p> <p>The receiver detects that the System Time from the signal-in-space time and the local timing realisation do not meet the synchronisation requirement and it stops processing OSNMA data as defined in OSNMA Receiver Guidelines.</p> <p>Pass/fail criteria: the receiver detects the replay and does not compute an authenticated valid position since the start of the replay until the end of the test.</p>	Appendix 12, GNS_3b

2.5	OSNMA hot start with false data	<p>Objective: Verify that OSNMA detects false data.</p> <p>Procedure: the GNSS receiver starts in GNSS and OSNMA hot start conditions.</p> <p>The GNSS receiver shall be able to acquire the signal of all the visible satellites and verify the authenticity of their navigation messages from OSNMA.</p> <p>At least one bit of the ephemeris data provided by each satellite does not correspond with the original and authenticated data, but the I/NAV message must be coherent, including CRC.</p> <p>Pass/fail criteria: the receiver detects the false data within 160 seconds and does not compute an authenticated valid position until the end of the test.</p>	Appendix 12, GNS_3b
-----	---------------------------------	---	---------------------

Sub-appendix 10. Security requirements

This **sub**-appendix specifies the IT security requirements for the smart tachograph system components (second-generation tachograph).

SEC_001 The following components of the smart tachograph system shall be security certified according to the Common Criteria scheme:

- vehicle unit
- tachograph card,
- motion sensor,
- external GNSS facility.

SEC_002 The minimum IT security requirements to be met by each component needing to be security certified shall be defined in a component Protection Profile, according to the Common Criteria scheme.

SEC_003 ~~The European Commission shall make sure that~~ **following** four Protection Profiles compliant with **this sub-appendix shall be** sponsored, developed, approved by the governmental IT security certification bodies organised within the Joint Interpretation Working Group (JIWG) which is supporting the mutual recognition of certificates under the umbrella of the European SOGIS-MRA (Agreement on Mutual Recognition of Information Technology Security Evaluation Certificates) and registered:

- Protection Profile for vehicle unit,
- Protection Profile for tachograph card,
- Protection Profile for motion sensor,
- Protection Profile for external GNSS facility.

The Protection Profile for vehicle unit shall address the cases when the VU is designed to be used or not with an external GNSS facility. In the former case, the security requirements of the external GNSS facility are provided in the dedicated Protection Profile.

SEC_004 Component manufacturers shall refine and complete the appropriate component Protection Profile as necessary, without amending or deleting existing threats, objectives, procedural means and security enforcing functions specifications, in order to build a Security Target against which they shall seek the security certification of the component.

SEC_005 Strict conformance of such specific Security Target with the corresponding Protection Profile must be stated during the evaluation process.

SEC_006 The assurance level for each Protection Profile shall be EAL4 augmented by the assurance components ATE_DPT.2 and AVA_VAN.5.

Sub-appendix 11. Common security mechanisms

TABLE OF CONTENTS

<i>Preamble</i>	423
PART A FIRST-GENERATION TACHOGRAPH SYSTEM	423
1. <i>Introduction</i>	423
1.1. References	423
1.2. Notations and abbreviated terms	424
2. <i>Cryptographic systems and algorithms</i>	425
2.1. Cryptographic systems	425
2.2. Cryptographic algorithms	425
2.2.1 RSA algorithm.....	425
2.2.2 Hash algorithm.....	426
2.2.3 Data Encryption Algorithm.....	426
3. <i>Keys and certificates</i>	426
3.1. Keys generation and distribution	426
3.1.1 RSA Keys generation and distribution.....	427
3.1.2 RSA Test keys.....	427
3.1.3 Motion sensor keys.....	428
3.1.4 T-DES session keys generation and distribution.....	428
3.2. Keys	428
3.3. Certificates	428
3.3.1 Certificates content.....	430
3.3.2 Certificates issued.....	431
3.3.3 Certificate verification and unwrapping.....	431
4. <i>Mutual authentication mechanism</i>	435
5. <i>VU-Cards data transfer confidentiality, integrity and authentication mechanisms</i>	435
5.1. Secure Messaging	436
5.2. Treatment of Secure Messaging errors	437
5.3. Algorithm to compute Cryptographic Checksums	438
5.4. Algorithm to compute cryptograms for confidentiality DOs	438
6. <i>Data download digital signature mechanisms</i>	438
6.1. Signature generation	440
6.2. Signature verification	440
PART B SECOND-GENERATION TACHOGRAPH SYSTEM	440
7. <i>Introduction</i>	440
7.1. References	440
7.2. Notations and Abbreviations	440
7.3. Definitions	442
8. <i>Cryptographic systems and algorithms</i>	442
8.1. Cryptographic Systems	443
8.2. Cryptographic Algorithms	443
8.2.1 Symmetric Algorithms.....	443
8.2.2 Asymmetric Algorithms and Standardized Domain Parameters.....	443
8.2.3 Hashing algorithms.....	443
8.2.4 Cipher Suites.....	444
9. <i>Keys and Certificates</i>	444
9.1. Asymmetric Key Pairs and Public Key Certificates	444
9.1.1 General.....	444

9.1.2	European Level	445
9.1.3	Member State Contracting Party Level	445
9.1.4	Equipment Level: Vehicle Units	447
9.1.5	Equipment Level: Tachograph Cards.....	448
9.1.6	Equipment Level: External GNSS Facilities.....	449
9.1.7	Overview: Certificate Replacement	450
9.2.	Symmetric Keys	452
9.2.1	Keys for Securing VU – Motion Sensor Communication	452
9.2.2	Keys for Securing DSRC Communication.....	452
9.3.	Certificates	459
9.3.1	General	459
9.3.2	Certificate Content.....	461
9.3.3	Requesting Certificates	462
10.	VU- Card Mutual Authentication and Secure Messaging	462
10.1.	General.....	463
10.2.	Mutual Certificate Chain Verification	463
10.2.1	Card Certificate Chain Verification by VU	466
10.2.2	VU Certificate Chain Verification by Card	469
10.3.	VU Authentication.....	470
10.4.	Chip Authentication and Session Key Agreement	472
10.5.	Secure Messaging.....	472
10.5.1	General	473
10.5.2	Secure Message Structure	476
10.5.3	Secure Messaging Session Abortion	477
11.	VU – External GNSS Facility Coupling, Mutual Authentication and Secure Messaging	477
11.1.	General.....	478
11.2.	VU and External GNSS Facility Coupling	478
11.3.	Mutual Certificate Chain Verification	478
11.3.1	General	478
11.3.2	During VU – EGF Coupling.....	478
11.3.3	During Normal Operation	479
11.4.	VU Authentication, Chip Authentication and Session Key Agreement.....	480
11.5.	Secure Messaging.....	480
12.	VU – Motion Sensor Pairing and Communication.....	481
12.1.	General.....	482
12.2.	VU – Motion Sensor Pairing Using Different Key Generations	484
12.3.	VU – Motion Sensor Pairing and Communication using AES	484
12.4.	VU – Motion Sensor Pairing For Different Equipment Generations	484
13.	Security for Remote Communication over DSRC.....	484
13.1.	General.....	484
13.2.	Tachograph Payload Encryption and MAC Generation	485
13.3.	Verification and Decryption of Tachograph Payload.....	486
14.	Signing Data Downloads and Verifying Signatures.....	487
14.1.	General.....	487
14.2.	Signature generation.....	487
14.3.	Signature verification	488

Preamble

This ~~Appendix~~ **sub-appendix** specifies the security mechanisms ensuring

- mutual authentication between different components of the tachograph system.
- confidentiality, integrity, authenticity and/or non-repudiation of data transferred between different components of the tachograph system or downloaded to external storage media.

This ~~Appendix~~ **sub-appendix** consists of two parts. Part A defines the security mechanisms for the first-generation tachograph system (digital tachograph). Part B defines the security mechanisms for the second-generation tachograph system (smart tachograph).

The mechanisms specified in Part A of this ~~Appendix~~ **sub-appendix** shall apply if at least one of the components of the tachograph system involved in a mutual authentication and/or data transfer process is of the first generation.

The mechanisms specified in Part B of this ~~Appendix~~ **sub-appendix** shall apply if both components of the tachograph system involved in the mutual authentication and/or data transfer process are of the second generation.

~~Appendix~~ **Sub-appendix** 15 provides more information regarding the use of first generation components in combination with second-generation components.

PART A FIRST-GENERATION TACHOGRAPH SYSTEM

1. Introduction

1.1. References

The following references are used in this ~~Appendix~~ **sub-appendix**:

- SHA-1 National Institute of Standards and Technology (NIST). *FIPS Publication 180-1 : Secure Hash Standard*. April 1995.
- PKCS1 RSA Laboratories. *PKCS # 1 : RSA Encryption Standard*. Version 2.0. October 1998.
- TDES National Institute of Standards and Technology (NIST). *FIPS Publication 46-3 : Data Encryption Standard*. Draft 1999.
- TDES-OP ANSI X9.52, Triple Data Encryption Algorithm Modes of Operation. 1998.
- ISO/IEC 7816-4 Information Technology – Identification cards – Integrated circuit(s) cards with contacts – Part 4: Interindustry commands for interexchange. First edition: 1995 + Amendment 1: 1997.
- ISO/IEC 7816-6 Information Technology – Identification cards – Integrated circuit(s) cards with contacts – Part 6: Interindustry data elements. First edition: 1996 + Cor 1: 1998.
- ISO/IEC 7816-8 Information Technology – Identification cards – Integrated circuit(s) cards with contacts – Part 8: Security related interindustry commands. First edition 1999.

- ISO/IEC 9796-2 Information Technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Mechanisms using a hash function. First edition: 1997.
- ISO/IEC 9798-3 Information Technology – Security techniques – Entity authentication mechanisms – Part 3: Entity authentication using a public key algorithm. Second edition 1998.
- ISO 16844-3 Road vehicles – Tachograph systems – Part 3: Motion sensor interface.

1.2. Notations and abbreviated terms

The following notations and abbreviated terms are used in this ~~Appendix~~ **sub-appendix**:

(K_a, K_b, K_c)	a key bundle for use by the Triple Data Encryption Algorithm,
CA	Certification Authority,
CAR	Certification Authority Reference,
CC	Cryptographic Checksum,
CG	Cryptogram,
CH	Command Header,
CHA	Certificate Holder Authorisation,
CHR	Certificate Holder Reference,
D()	Decryption with DES,
DE	Data Element,
DO	Data Object,
d	RSA private key, private exponent,
e	RSA public key, public exponent,
E()	Encryption with DES,
EQT	Equipment,
$Hash()$	hash value, an output of $Hash$,
$Hash$	hash function,
KID	Key Identifier,
K_m	TDES key. Master Key defined in ISO 16844-3.
K_{mVU}	TDES key inserted in vehicle units.
K_{mWC}	TDES key inserted in workshop cards.
m	message representative, an integer between 0 and $n-1$,
n	RSA keys, modulus,
PB	Padding Bytes,
PI	Padding Indicator byte (for use in Cryptogram for confidentiality DO),
PV	Plain Value,
s	signature representative, an integer between 0 and $n-1$,

SSC	Send Sequence Counter,
SM	Secure Messaging,
TCBC	TDEA Cipher Block Chaining Mode of Operation
TDEA	Triple Data Encryption Algorithm,
TLV	Tag Length Value,
VU	Vehicle Unit,
X.C	the certificate of user X issued by a certification authority,
X.CA	a certification authority of user X,
X.CA.PK _o X.C	the operation of unwrapping a certificate to extract a public key. It is an infix operator, whose left operand is the public key of a certification authority, and whose right operand is the certificate issued by that certification authority. The outcome is the public key of the user X whose certificate is the right operand,
X.PK	RSA public key of a user X,
X.PK[I] of user X,	RSA encipherment of some information I, using the public key of user X,
X.SK	RSA private key of a user X,
X.SK[I] of user X,	RSA encipherment of some information I, using the private key of user X,
'xx'	an Hexadecimal value,
	concatenation operator.

2. Cryptographic systems and algorithms

2.1. Cryptographic systems

CSM_001 Vehicle units and tachograph cards shall use a classical RSA public-key cryptographic system to provide the following security mechanisms:

- authentication between vehicle units and cards,
- transport of Triple-DES session keys between vehicle units and tachograph cards,
- digital signature of data downloaded from vehicle units or tachograph cards to external media.

CSM_002 Vehicle units and tachograph cards shall use a Triple DES symmetric cryptographic system to provide a mechanism for data integrity during user data exchange between vehicle units and tachograph cards, and to provide, where applicable, confidentiality of data exchange between vehicle units and tachograph cards.

2.2. Cryptographic algorithms

2.2.1 RSA algorithm

CSM_003 The RSA algorithm is fully defined by the following relations:

$$X.SK[m] = s = m^d \text{ mod } n$$

$$X.PK[s] = m = s^e \text{ mod } n$$

A more comprehensive description of the RSA function can be found in reference [PKCS1]. Public exponent, e, for RSA calculations is an integer between 3 and n-1 satisfying $\text{gcd}(e, \text{lcm}(p-1, q-1))=1$.

2.2.2 Hash algorithm

CSM_004 The digital signature mechanisms shall use the SHA-1 hash algorithm as defined in reference [SHA-1].

2.2.3 Data Encryption Algorithm

CSM_005 DES based algorithms shall be used in Cipher Block Chaining mode of operation.

3. Keys and certificates

3.1. Keys generation and distribution

3.1.1 RSA Keys generation and distribution

CSM_006 RSA keys shall be generated through three functional hierarchical levels:

- ~~European Root~~ level,
- ~~Member State Contracting Party~~ level,
- Equipment level.

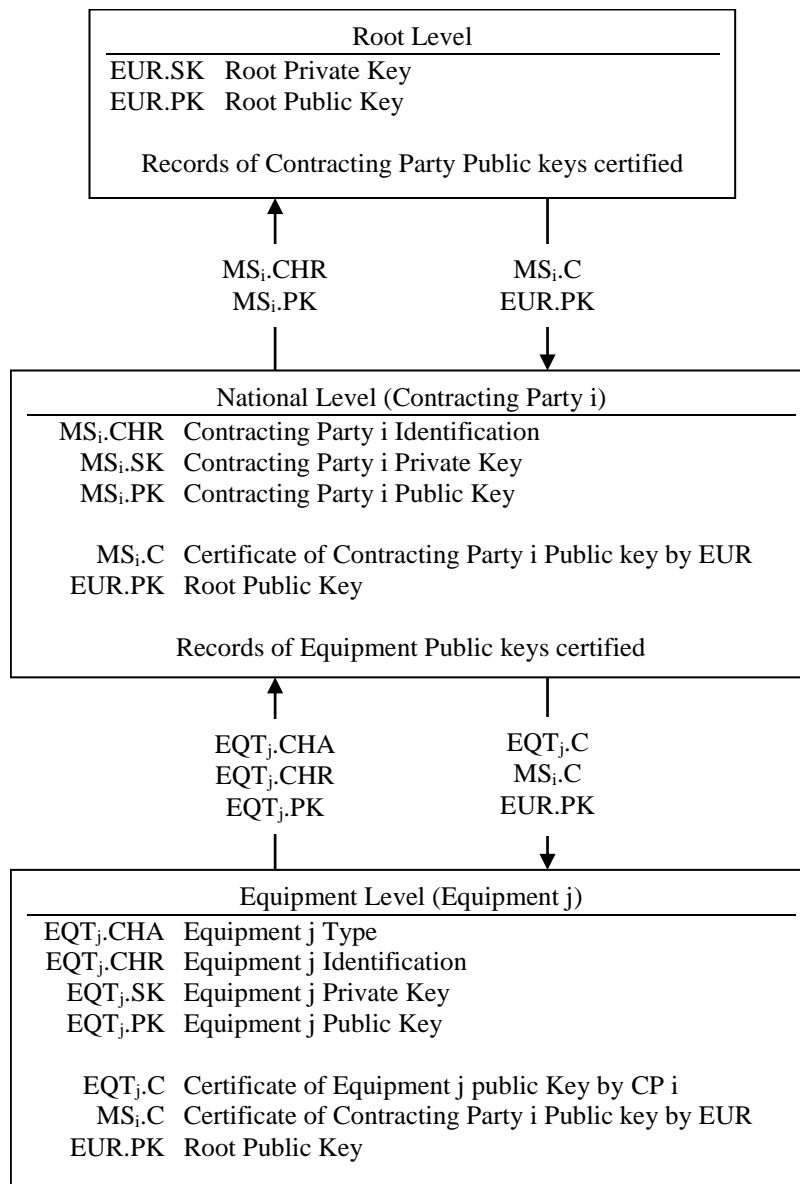
CSM_007 At ~~European root~~ level, a single ~~European root~~ key pair (EUR.SK and EUR.PK) shall be generated. The ~~European root~~ private key shall be used to certify the ~~Member State Contracting Parties~~ public keys. Records of all certified keys shall be kept. These tasks shall be handled by a ~~European Root~~ Certification Authority, ~~under the authority and responsibility of the European Commission.~~

CSM_008 At ~~Member State Contracting Party~~ level, a ~~Member State Contracting Party~~ key pair (MS.SK and MS.PK) shall be generated. ~~Member States Contracting Parties~~ public keys shall be certified by the ~~European Root~~ Certification Authority. The ~~Member State Contracting Party~~ private key shall be used to certify public keys to be inserted in equipment (vehicle unit or tachograph card). Records of all certified public keys shall be kept with the identification of the equipment to which it is intended. These tasks shall be handled by a ~~Member State Contracting Party~~ Certification Authority. A ~~Member State Contracting Party~~ may regularly change its key pair.

CSM_009 At equipment level, one single key pair (EQT.SK and EQT.PK) shall be generated and inserted in each equipment. Equipment public keys shall be certified by a ~~Member State Contracting Party~~ Certification Authority. These tasks may be handled by equipment manufacturers, equipment personalisers or ~~Member State Contracting Party~~ authorities. This key pair is used for authentication, digital signature and encipherment services

CSM_010 Private keys confidentiality shall be maintained during generation, transport (if any) and storage.

The following picture summarises the data flow of this process:



3.1.2 RSA Test keys

CSM_011 For the purpose of equipment testing (including interoperability tests) the ~~European~~ **Root Certification Authority** shall generate a different single ~~European~~ **root** test key pair and at least two ~~Member State~~ **Contracting Party** test key pairs, the public keys of which shall be certified with the ~~European~~ **root** private test key. Manufacturers shall insert, in equipment undergoing type approval tests, test keys certified by one of these ~~Member State~~ **Contracting Party** test keys.

3.1.3 Motion sensor keys

The confidentiality of the three Triple DES keys described below shall be appropriately maintained during generation, transport (if any) and storage.

In order to support tachograph components compliant with ISO 16844, the ~~European~~ **Root** Certification Authority and the ~~Member State~~ **Contracting Party** Certification Authorities shall, in addition, ensure the following:

CSM_036 The ~~European~~ **Root** Certification authority shall generate $K_{m_{VU}}$ and $K_{m_{WC}}$, two independent and unique Triple DES keys, and generate K_m as: $K_m = K_{m_{VU}} \text{ XOR } K_{m_{WC}}$. The ~~European~~ **Root** Certification Authority shall forward these keys, under appropriately secured procedures, to ~~Member States~~ **Contracting Parties** Certification Authorities at their request.

CSM_037 ~~Member States~~ **Contracting Parties** Certification Authorities shall:

- use K_m to encrypt motion sensor data requested by motion sensor manufacturers (data to be encrypted with K_m is defined in ISO 16844-3),
- forward $K_{m_{VU}}$ to vehicle unit manufacturers, under appropriately secured procedures, for insertion in vehicle units,
- ensure that $K_{m_{WC}}$ will be inserted in all workshop cards (SensorInstallationSecData in Sensor_Installation_Data elementary file) during card personalisation.

3.1.4 T-DES session keys generation and distribution

CSM_012 Vehicle units and tachograph cards shall, as a part of the mutual authentication process, generate and exchange necessary data to elaborate a common Triple DES session key. This exchange of data shall be protected for confidentiality through an RSA crypt-mechanism.

CSM_013 This key shall be used for all subsequent cryptographic operations using secure messaging. Its validity shall expire at the end of the session (withdrawal of the card or reset of the card) and/or after 240 use (one use of the key = one command using secure messaging sent to the card and associated response).

3.2. Keys

CSM_014 RSA keys shall have (whatever the level) the following lengths: modulus n 1024 bits, public exponent e 64 bits maximum, private exponent d 1024 bits.

CSM_015 Triple DES keys shall have the form (K_a, K_b, K_a) where K_a and K_b are independent 64 bits long keys. No parity error detecting bits shall be set.

3.3. Certificates

CSM_016 RSA Public key certificates shall be “non self-descriptive” “Card Verifiable” certificates (Ref.: ISO/IEC 7816-8)

3.3.1 Certificates content

CSM_017 RSA Public key certificates are built with the following data in the following order:

<i>Data</i>	<i>Format</i>	<i>Bytes</i>	<i>Obs</i>
CPI	INTEGER	1	Certificate Profile Identifier ('01' for this version)
CAR	OCTET STRING	8	Certification Authority Reference
CHA	OCTET STRING	7	Certificate Holder Authorisation
EOV	TimeReal	4	Certificate end of validity. Optional, "FF" padded if not used.
CHR	OCTET STRING	8	Certificate Holder Reference
<i>n</i>	OCTET STRING	128	Public key (modulus)
<i>e</i>	OCTET STRING	8	Public Key (public exponent)
164			

Notes:

1. The "Certificate Profile Identifier" (CPI) delineates the exact structure of an authentication certificate. It can be used as an equipment internal identifier of a relevant headerlist which describes the concatenation of Data Elements within the certificate.

The headerlist associated with this certificate content is as follows:

	'4D'	'16'	'5F 29'	'01'	'42'	'08'	'5F 4B'	'07'	'5F 24'	'04'	'5F 20'	'08'	'7F 49'	'05'	'81'	'81 80'	'82'	'08'
Extended Headerlist Tag																		
Length of header list																		
			CPI Tag	CPI Length	CAR Tag	CAR Length	CHA Tag	CHA Length	EOV Tag	EOV Length	CHR Tag	CHR Length	Public Key Tag (Constructed)	Length of subsequent DOs	modulus Tag	modulus length	public exponent Tag	public exponent length

2. The "Certification Authority Reference" (CAR) has the purpose of identifying the certificate issuing CA, in such a way that the Data Element can be used at the same time as an Authority Key Identifier to reference the Public Key of the Certification Authority (for coding, see Key Identifier below).

3. The "Certificate Holder Authorisation" (CHA) is used to identify the rights of the certificate holder. It consists of the Tachograph Application ID and of the type of equipment to which the certificate is intended (according to EquipmentType data element, '00' for a ~~Member State Contracting Party~~).

4. The "Certificate Holder Reference" (CHR) has the purpose of identifying uniquely the certificate holder, in such a way that the Data Element can be used at the same time as a Subject Key Identifier to reference the Public Key of the certificate holder.

5. Key Identifiers uniquely identify certificate holder or certification authorities. They are coded as follows:

5.1. Equipment (VU or Card):

<i>Data</i>	<i>Equipment serial number</i>	<i>Date</i>	<i>Type</i>	<i>Manufacturer</i>
Length	4 Bytes	2 Bytes	1 Byte	1 Byte
Value	Integer	mm yy BCD coding	Manufacturer specific	Manufacturer code

In the case of a VU, the manufacturer, when requesting certificates, may or may not know the identification of the equipment in which the keys will be inserted.

In the first case, the manufacturer will send the equipment identification with the public key to its ~~Member State Contracting Party~~ authority for certification. The certificate will then contain the equipment identification, and the manufacturer must ensure that keys and certificate are inserted in the intended equipment. The Key identifier has the form shown above.

In the later case, the manufacturer must uniquely identify each certificate request and send this identification with the public key to its ~~Member State Contracting Party~~ authority for certification. The certificate will contain the request identification. The manufacturer must feed back its ~~Member State Contracting Party~~ authority with the assignment of key to equipment (i.e. certificate request identification, equipment identification) after key installation in the equipment. The key identifier has the following form:

<i>Data</i>	<i>Certificate request serial number</i>	<i>Date</i>	<i>Type</i>	<i>Manufacturer</i>
Length	4 Bytes	2 Bytes	1 Byte	1 Byte
Value	Integer	mm yy BCD coding	'FF'	Manufacturer code

5.2. Certification Authority:

<i>Data</i>	<i>Authority Identification</i>	<i>Key serial number</i>	<i>Additional info</i>	<i>Identifier</i>
Length	4 Bytes	1 Byte	2 Bytes	1 Byte
Value	1 Byte nation numerical code 3 Bytes nation alphanumerical code	Integer	additional coding (CA specific) 'FF FF' if not used	'01'

The key serial number is used to distinguish the different keys of a ~~Member State Contracting Party~~, in the case the key is changed.

6. Certificate verifiers shall implicitly know that the public key certified is an RSA key relevant to authentication, digital signature verification and encipherement for confidentiality services (the certificate contains no Object Identifier to specify it).

3.3.2 Certificates issued

CSM_018 The certificate issued is a digital signature with partial recovery of the certificate content in accordance with ISO/IEC 9796-2 (except for its annex A4), with the "Certification Authority Reference" appended.

$$X.C = X.CA.SK['6A' || C_r || Hash(Cc) || 'BC'] || C_n || X.CAR$$

With certificate content = Cc = C_r || C_n
 106 bytes 58 bytes

Notes:

1. This certificate is 194 bytes long.
2. CAR, being hidden by the signature, is also appended to the signature, such that the Public Key of the Certification Authority may be selected for the verification of the certificate.
3. The certificate verifier shall implicitly know the algorithm used by the Certification Authority to sign the certificate.

4. The header list associated with this issued certificate is as follows:

'7F 21'	'09'	'5F 37'	'81 80'	'5F 38'	'3A'	'42'	'08'
CV Certificate Tag (Constructed)	Length of subsequent DOs	Signature Tag	Signature Length	Remainder Tag	Remainder Length	CAR Tag	CAR Length

3.3.3 Certificate verification and unwrapping

Certificate verification and unwrapping consists in verifying the signature in accordance with ISO/IEC 9796-2, retrieving the certificate content and the public key contained: X.PK = X.CA.PK ◦ X.C, and verifying the validity of the certificate.

CSM_019 It involves the following steps:

Verify signature and retrieve content:

- from X.C retrieve Sign, C_n' and CAR': X.C = Sign || C_n' || CAR'
 128 Bytes 58 Bytes 8 Bytes

from CAR' select appropriate Certification Authority Public Key (if not done before through other means)

- open Sign with CA Public Key: Sr' = X.CA.PK [Sign],
- check Sr' starts with '6A' and ends with 'BC'
- compute C_r' and H' from: Sr' = '6A' || C_r' || H' || 'BC'
 106 Bytes 20 Bytes
- Recover certificate content C' = C_r' || C_n' ,
- check Hash(C') = H'

If the checks are OK the certificate is a genuine one, its content is C'.

Verify validity. From C':

- if applicable, check End of validity date,

Retrieve and store public key, Key Identifier, Certificate Holder Authorisation and Certificate End of Validity from C':

- X.PK = n || e
- X.KID = CHR
- X.CHA = CHA
- X.EOV = EOVS

4. Mutual authentication mechanism

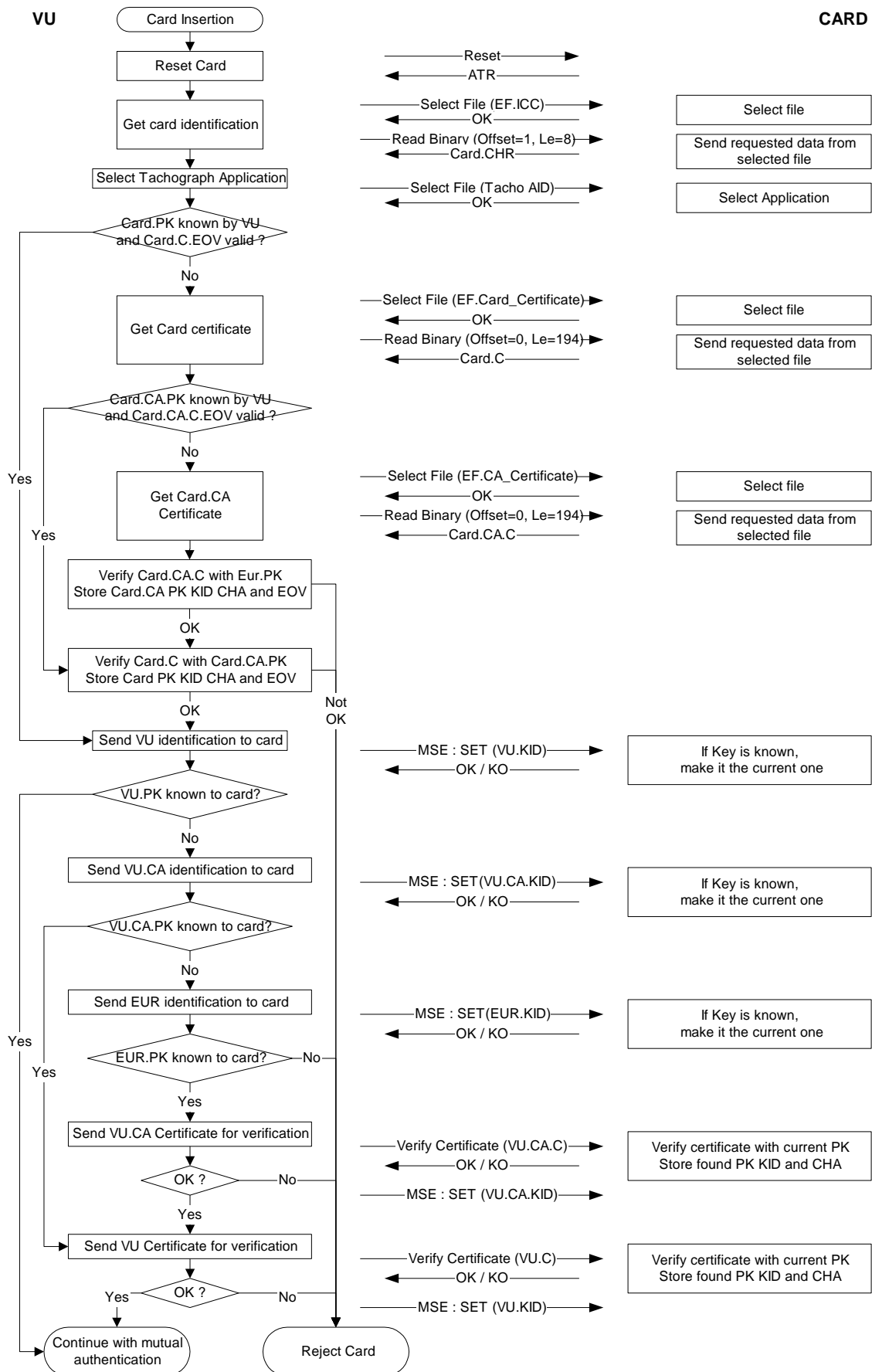
Mutual authentication between cards and VUs is based on the following principle:

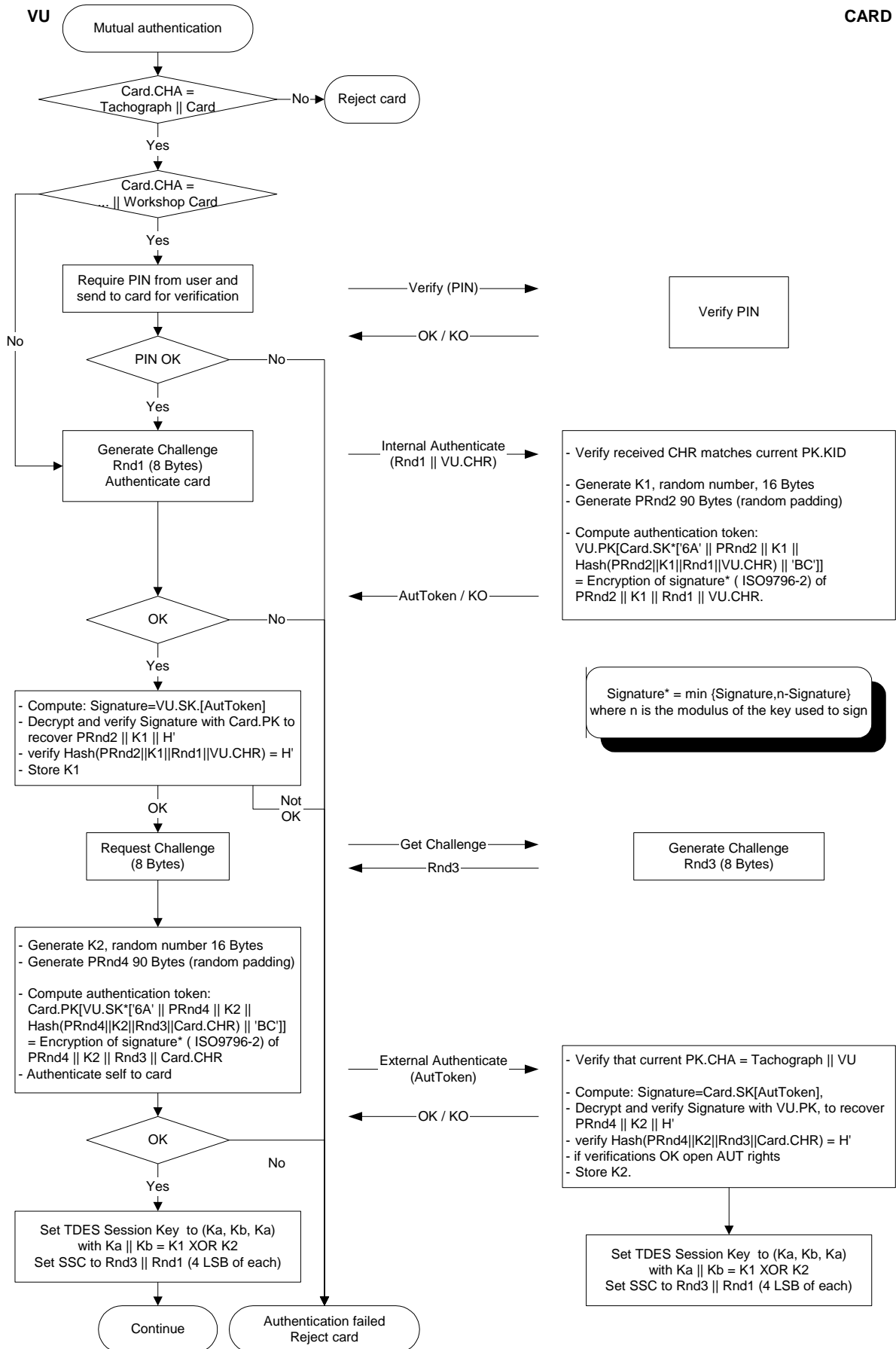
Each party shall demonstrate to the other that it owns a valid key pair, the public key of which has been certified by a ~~Member State~~ **Contracting Party** certification authority, itself being certified by the ~~European certification authority~~ **Root Certification Authority**.

Demonstration is made by signing with the private key a random number sent by the other party, who must recover the random number sent when verifying this signature.

The mechanism is triggered at card insertion by the VU. It starts with the exchange of certificates and unwrapping of public keys, and ends with the setting of a session key.

CSM_020 The following protocol shall be used (arrows indicate commands and data exchanged (see ~~Appendix~~ **Sub-appendix 2**)):





5. VU-Cards data transfer confidentiality, integrity and authentication mechanisms

5.1. Secure Messaging

CSM_021 VU-Cards data transfers integrity shall be protected through Secure Messaging in accordance with references [ISO/IEC 7816-4] and [ISO/IEC 7816-8].

CSM_022 When data need to be protected during transfer, a Cryptographic Checksum Data Object shall be appended to the Data Objects sent within the command or the response. The Cryptographic Checksum shall be verified by the receiver.

CSM_023 The cryptographic checksum of data sent within a command shall integrate the command header, and all data objects sent (\Rightarrow CLA = '0C', and all data objects shall be encapsulated with tags in which b1=1).

CSM_024 The response status-information bytes shall be protected by a cryptographic checksum when the response contains no data field.

CSM_025 Cryptographic checksums shall be 4 Bytes long.

The structure of commands and responses when using secure messaging is therefore the following:

The DOs used are a partial set of the Secure Messaging DOs described in ISO/IEC 7816-4:

<i>Tag</i>	<i>Mnemonic</i>	<i>Meaning</i>
'81'	T _{PV}	Plain Value not BER-TLV coded data (to be protected by CC)
'97'	T _{LE}	Value of Le in the unsecured command (to be protected by CC)
'99'	T _{SW}	Status-Info (to be protected by CC)
'8E'	T _{CC}	Cryptographic Checksum
'87'	T _{PI CG}	Padding Indicator Byte Cryptogram (Plain Value not coded in BER-TLV)

Given an unsecured command response pair:

<i>Command header</i>				<i>Command body</i>		
CLA	INS	P1	P2	[L _c field]	[Data field]	[L _c field]
four bytes				L bytes, denoted as B ₁ to B _L		

<i>Response body</i>	<i>Response trailer</i>
[Data field]	SW1 SW2
L _r data bytes	two bytes

The corresponding secured command response pair is:

Secured command:

Command header (CH)				Command body										
CLA	INS	P1	P2	[New L _c field]	[New Data field]						[New L _e field]			
'OC'				Length of New Data field	T _{PV}	L _{PV}	PV	T _{LE}	L _{LE}	L _e	T _{CC}	L _{CC}	CC	'00'
					'81'	L _c	Data field	'97'	'01'	L _e	'8E'	'04'	CC	

Data to be integrated in checksum = CH || PB || T_{PV} || L_{PV} || PV || T_{LE} || L_{LE} || L_e || PB

PB = Padding Bytes (80 .. 00) in accordance with ISO-IEC 7816-4 and ISO 9797 method 2.

DOs PV and LE are present only when there is some corresponding data in the unsecured command.

Secured response:

- Case where response data field is not empty and needs not to be protected for confidentiality:

Response body						Response trailer
[New Data field]				new SW1 SW2		
T _{PV}	L _{PV}	PV	T _{CC}	L _{CC}	CC	
'81'	L _r	Data field	'8E'	'04'	CC	

Data to be integrated in checksum = T_{PV} || L_{PV} || PV || PB

- Case where response data field is not empty and needs to be protected for confidentiality:

Response body						Response trailer
[New Data field]				new SW1 SW2		
T _{PI CG}	L _{PI CG}	PI CG	T _{CC}	L _{CC}	CC	
'87'		PI CG	'8E'	'04'	CC	

Data to be carried by CG: non BER-TLV coded data and padding bytes.

Data to be integrated in checksum = T_{PI CG} || L_{PI CG} || PI CG || PB

- Case where response data field is empty:

Response body						Response trailer
[New Data field]				new SW1 SW2		
T _{SW}	L _{SW}	SW	T _{CC}	L _{CC}	CC	
'99'	'02'	New SW1 SW2	'8E'	'04'	CC	

Data to be integrated in checksum = T_{SW} || L_{SW} || SW || PB

5.2. Treatment of Secure Messaging errors

CSM_026 When the tachograph card recognises an SM error while interpreting a command, then the status bytes must be returned without SM. In accordance with ISO/IEC 7816-4, the following status bytes are defined to indicate SM errors:

'66 88': Verification of Cryptographic Checksum failed,

'69 87': Expected SM Data Objects missing,

'69 88': SM Data Objects incorrect.

CSM_027 When the tachograph card returns status bytes without SM DOs or with an erroneous SM DO, the session must be aborted by the VU.

5.3. Algorithm to compute Cryptographic Checksums

CSM_028 Cryptographic checksums are built using a retail MACs in accordance with ANSI X9.19 with DES:

- Initial stage: The initial check block y_0 is $E(K_a, SSC)$.
- Sequential stage: The check blocks y_1, \dots, y_n are calculated using K_a .
- Final stage: The cryptographic checksum is calculated from the last check block y_n as follows: $E(K_a, D(K_b, y_n))$.

where $E()$ means encryption with DES, and $D()$ means decryption with DES.

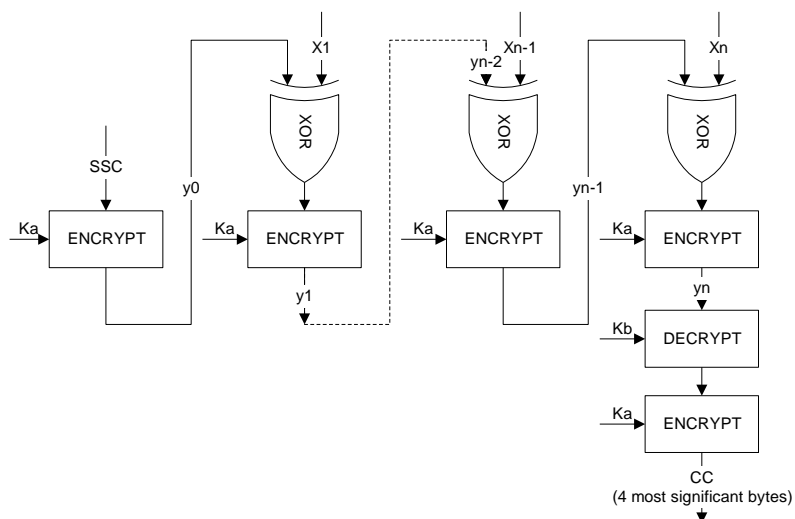
The four most significant bytes of the cryptographic checksum are transferred

CSM_029 The Send Sequence Counter (SSC) shall be initiated during key agreement procedure to:

Initial SSC: Rnd3 (4 least significant bytes) || Rnd1 (4 least significant bytes).

CSM_030 The Send Sequence Counter shall be increased by 1 each time before a MAC is calculated (i.e. the SSC for the first command is Initial SSC + 1, the SSC for the first response is Initial SSC + 2).

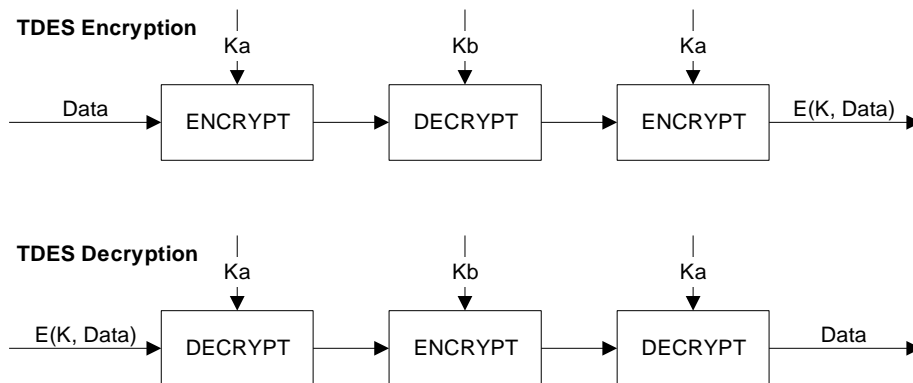
The following figure shows the calculation of the retail MAC:



5.4. Algorithm to compute cryptograms for confidentiality DOs

CSM_031 Cryptograms are computed using TDEA in TCBC mode of operation in accordance with references [TDES] and [TDES-OP] and with the Null vector as Initial Value block.

The following figure shows the application of keys in TDES:



6. Data download digital signature mechanisms

CSM_032 The Intelligent Dedicated Equipment (IDE) stores data received from an equipment (VU or card) during one download session within one physical data file. This file must contain the certificates MS_i.C and EQT.C. The file contains digital signatures of data blocks as specified in **Appendix Sub-appendix 7 Data Downloading Protocols**.

CSM_033 Digital signatures of downloaded data shall use a digital signature scheme with appendix such, that downloaded data may be read without any decipherment if desired.

6.1. Signature generation

CSM_034 Data signature generation by the equipment shall follow the signature scheme with appendix defined in reference [PKCS1] with the SHA-1 hash function :

Signature = EQT.SK['00' || '01' || PS || '00' || DER(SHA-1(Data))]

PS = Padding string of octets with value 'FF' such that length is 128.

DER(SHA-1(M)) is the encoding of the algorithm ID for the hash function and the hash value into an ASN.1 value of type `DigestInfo` (distinguished encoding rules):

'30' || '21' || '30' || '09' || '06' || '05' || '2B' || '0E' || '03' || '02' || '1A' || '05' || '00' || '04' || '14' || Hash Value.

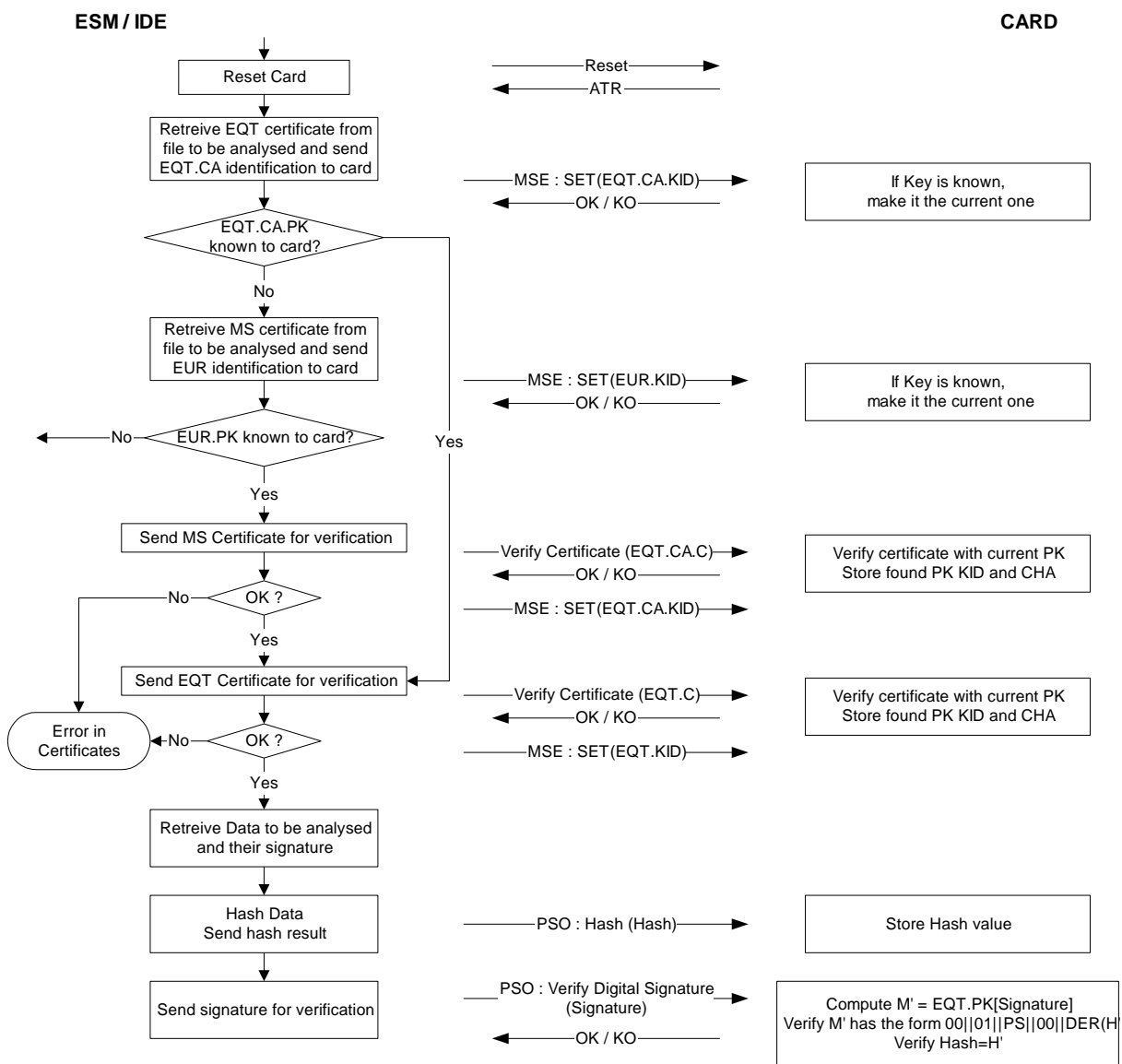
6.2. Signature verification

CSM_035 Data signature verification on downloaded data shall follow the signature scheme with appendix defined in reference [PKCS1] with the SHA-1 hash function.

The ~~European~~ **Root** public key EUR.PK needs to be known independently (and trusted) by the verifier.

The following table illustrates the protocol an IDE carrying a Control card can follow to verify the integrity of data downloaded and stored on the ESM (External Storage media). The control card is used to perform the decipherment of digital signatures. This function may in this case not be implemented in the IDE.

The equipment that has downloaded and signed the data to be analysed is denoted EQT.



PART B SECOND-GENERATION TACHOGRAPH SYSTEM

7. Introduction

7.1. References

The following references are used in this part of this ~~Appendix~~ **sub-appendix**.

AES National Institute of Standards and Technology (NIST), FIPS PUB 197: Advanced Encryption Standard (AES), November 26, 2001

DSS National Institute of Standards and Technology (NIST), FIPS PUB 186-4: Digital Signature Standard (DSS), July 2013

ISO 7816-4 ISO/IEC 7816-4, Identification cards - Integrated circuit cards - Part 4: Organization, security and commands for interchange. Third edition 2013-04-15

ISO 7816-8 ISO/IEC 7816-8, Identification cards - Integrated circuit cards - Part 8: Commands for security operations. Second edition 2004-06-01

ISO 8825-1 ISO/IEC 8825-1, Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER). Fourth edition, 2008-12-15

ISO 9797-1 ISO/IEC 9797-1, Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher. Second edition, 2011-03-01

ISO 10116 ISO/IEC 10116, Information technology – Security techniques – Modes of operation of an n -bit block cipher. Third edition, 2006-02-01

ISO 16844-3 ISO/IEC 16844-3, Road vehicles – Tachograph systems – Part 3: Motion sensor interface. First edition 2004, including Technical Corrigendum 1 2006

RFC 5480 Elliptic Curve Cryptography Subject Public Key Information, March 2009

RFC 5639 Elliptic Curve Cryptography (ECC) - Brainpool Standard Curves and Curve Generation, 2010

RFC 5869 HMAC-based Extract-and-Expand Key Derivation Function (HKDF), May 2010

SHS National Institute of Standards and Technology (NIST), FIPS PUB 180-4: Secure Hash Standard, March 2012

SP 800-38B National Institute of Standards and Technology (NIST), Special Publication 800-38B: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, 2005

TR-03111 BSI Technical Guideline TR-03111, Elliptic Curve Cryptography, version 2.00, 2012-06-28

7.2. Notations and Abbreviations

The following notations and abbreviated terms are used in this ~~Appendix~~ **sub-appendix**:

AES Advanced Encryption Standard

CA	Certificate Authority
CAR	Certificate Authority Reference
CBC	Cipher Block Chaining (mode of operation)
CH	Command Header
CHA	Certificate Holder Authorisation
CHR	Certificate Holder Reference
CV	Constant Vector
DER	Distinguished Encoding Rules
DO	Data Object
DSRC	Dedicated Short Range Communication
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ECDH	Elliptic Curve Diffie-Hellman (key agreement algorithm)
EGF	External GNSS Facility
EQT	Equipment
IDE	Intelligent Dedicated Equipment
K_M	Motion Sensor Master Key, allowing the pairing of a Vehicle Unit to a Motion Sensor
K_{M-VU}	Key inserted in vehicle units, allowing a VU to derive the Motion Sensor Master Key if a workshop card is inserted into the VU
K_{M-WC}	Key inserted in workshop cards, allowing a VU to derive the Motion Sensor Master Key if a workshop card is inserted into the VU
MAC	Message Authentication Code
MoS	Motion Sensor
MSB	Most Significant Bit
PKI	Public Key Infrastructure
RCF	Remote Communication Facility
SSC	Send Sequence Counter
SM	Secure Messaging
TDES	Triple Data Encryption Standard
TLV	Tag Length Value
VU	Vehicle Unit
X.C	the public key certificate of user X
X.CA	the certificate authority that issued the certificate of user X
X.CAR	the certificate authority reference mentioned in the certificate of user X
X.CHR	the certificate holder reference mentioned in the certificate of user X
X.PK	public key of user X

X.SK	private key of user X
X.PK _{eph}	ephemeral public key of user X
X.SK _{eph}	ephemeral private key of user X
'xx'	a hexadecimal value
	concatenation operator

7.3. Definitions

The definitions of terms used in this ~~Appendix~~ **sub-appendix** are included in section I of ~~Annex~~ **Appendix 1C**.

8. Cryptographic systems and algorithms

8.1. Cryptographic Systems

CSM_38 Vehicle units and tachograph cards shall use an elliptic curve-based public-key cryptographic system to provide the following security services: mutual authentication between a vehicle unit and a card,

agreement of AES session keys between a vehicle unit and a card,

ensuring the authenticity, integrity and non-repudiation of data downloaded from vehicle units or tachograph cards to external media.

CSM_39 Vehicle units and external GNSS facilities shall use an elliptic curve-based public-key cryptographic system to provide the following security services:

- coupling of a vehicle unit and an external GNSS facility,
- mutual authentication between a vehicle unit and an external GNSS facility,
- agreement of an AES session key between a vehicle unit and an external GNSS facility.

CSM_40 Vehicle units and tachograph cards shall use an AES-based symmetric cryptographic system to provide the following security services:

- ensuring authenticity and integrity of data exchanged between a vehicle unit and a tachograph card,
- where applicable, ensuring confidentiality of data exchanged between a vehicle unit and a tachograph card.

CSM_41 Vehicle units and external GNSS facilities shall use an AES-based symmetric cryptographic system to provide the following security services:

- ensuring authenticity and integrity of data exchanged between a vehicle unit and an external GNSS facility.

CSM_42 Vehicle units and motion sensors shall use an AES-based symmetric cryptographic system to provide the following security services:

- pairing of a vehicle unit and a motion sensor,
- mutual authentication between a vehicle unit and a motion sensor,

- ensuring confidentiality of data exchanged between a vehicle unit and a motion sensor.

CSM_43 Vehicle units and control cards shall use an AES-based symmetric cryptographic system to provide the following security services on the remote communication interface:

- ensuring confidentiality, authenticity and integrity of data transmitted from a vehicle unit to a control card.

Notes:

- Properly speaking, data is transmitted from a vehicle unit to a remote interrogator under the control of a control officer, using a remote communication facility that may be internal or external to the VU, see ~~Appendix~~ **Sub-appendix** 14. However, the remote interrogator sends the received data to a control card for decryption and validation of authenticity. From a security point of view, the remote communication facility and the remote interrogator are fully transparent.
- A workshop card offers the same security services for the DSRC interface as a control card does. This allows a workshop to validate the proper functioning of the remote communication interface of a VU, including security. Please refer to section ~~9.2.3~~ **9.2.2** for more information.

8.2. Cryptographic Algorithms

8.2.1 Symmetric Algorithms

CSM_44 Vehicle units, tachograph cards, motion sensors and external GNSS facilities shall support the AES algorithm as defined in [AES], with key lengths of 128, 192 and 256 bits.

8.2.2 Asymmetric Algorithms and Standardized Domain Parameters

CSM_45 Vehicle units, tachograph cards and external GNSS facilities shall support elliptic curve cryptography with a key size of 256, 384 and 512/521 bits.

CSM_46 Vehicle units, tachograph cards and external GNSS facilities shall support the ECDSA signing algorithm, as specified in [DSS].

CSM_47 Vehicle units, tachograph cards and external GNSS facilities shall support the ECKA-EG key agreement algorithm, as specified in [TR 03111].

CSM_48 Vehicle units, tachograph cards and external GNSS facilities shall support all standardized domain parameters specified in **Table 43** below for elliptic curve cryptography.

Table 1

Standardized domain parameters

<i>Name</i>	<i>Size (bits)</i>	<i>Reference</i>	<i>Object identifier</i>
NIST P-256	256	[DSS], [RFC 5480]	secp256r1
BrainpoolP256r1	256	[RFC 5639]	brainpoolP256r1
NIST P-384	384	[DSS], [RFC 5480]	secp384r1
BrainpoolP384r1	384	[RFC 5639]	brainpoolP384r1
BrainpoolP512r1	512	[RFC 5639]	brainpoolP512r1
NIST P-521	521	[DSS], [RFC 5480]	secp521r1

Note: the object identifiers mentioned in the last column of ~~Table 43~~ **Table 1** are specified in [RFC 5639] for the Brainpool curves and in [RFC 5480] for the NIST curves.

Example 1: the object identifier of the BrainpoolP256r1 curve is {iso(1) identified-organization(3) teletrust(36) algorithm(3) signaturealgorithm(3) ecSign(2) ecStdCurvesAndGeneration(8) ellipticCurve(1) versionOne(1) 7}.

Or in dot notation: 1.3.36.3.3.2.8.1.1.7.

Example 2: the object identifier of the NIST P-384 curve is

{iso(1) identified-organization(3) certicom(132) curve(0) 34}.

Or in dot notation: 1.3.132.0.34.

8.2.3 Hashing algorithms

CSM_49 Vehicle units ~~and~~, tachograph cards **and external GNSS facilities** shall support the SHA-256, SHA-384 and SHA-512 algorithms specified in [SHS].

8.2.4 Cipher Suites

CSM_50 In case a symmetric algorithm, an asymmetric algorithm and/or a hashing algorithm are used together to form a security protocol, their respective key lengths and hash sizes shall be of (roughly) equal strength. ~~Table 44~~ **Table 2** shows the allowed cipher suites:

Table 2

Allowed cipher suites

<i>Cipher suite Id</i>	<i>ECC key size (bits)</i>	<i>AES key length (bits)</i>	<i>Hashing algorithm</i>	<i>MAC length (bytes)</i>
CS#1	256	128	SHA-256	8
CS#2	384	192	SHA-384	12
CS#3	512/521	256	SHA-512	16

Note: ECC keys sizes of 512 bits and 521 bits are considered to be equal in strength for all purposes within this ~~Appendix~~ **sub-appendix**.

9. Keys and Certificates

9.1. Asymmetric Key Pairs and Public Key Certificates

9.1.1 General

Note: the keys described in this section are used for mutual authentication and secure messaging between vehicle units and tachograph cards and between vehicle units and external GNSS facilities. These processes are described in detail in chapters 0 and 0 of this **sub-appendix**.

CSM_51 Within the ~~European~~ Smart Tachograph system, ECC key pairs and corresponding certificates shall be generated and managed through three functional hierarchical levels:

- ~~European~~ **Root** level,
- ~~Member State~~ **Contracting Party** level,
- Equipment level.

CSM_52 Within the entire ~~European~~ Smart Tachograph system, public and private keys and certificates shall be generated, managed and communicated using standardized and secure methods.

9.1.2 ~~European~~Root Level

CSM_53 At ~~European~~ root level, a single unique ECC key pair designated as EUR shall be generated. It shall consist of a private key (EUR.SK) and a public key (EUR.PK). This key pair shall form the root key pair of the entire ~~European~~ Smart Tachograph PKI. This task shall be handled by a ~~European~~ Root Certificate Authority (ERCA), ~~under the authority and the responsibility of the European Commission.~~

CSM_54 The ERCA shall use the ~~European~~ Root private key to sign a (self-signed) root certificate of the ~~European~~ Root public key, and shall communicate this ~~European~~ root certificate to all ~~Member States~~ **Contracting Parties**.

CSM_55 The ERCA shall use the ~~European~~ Root private key to sign the certificates of the ~~Member States~~ **Contracting Parties** public keys upon request. The ERCA shall keep records of all signed ~~Member State~~ **Contracting Party** public key certificates.

CSM_56 As shown in Figure 1 in section 9.1.7, the ERCA shall generate a new ~~European~~ root key pair every 17 years. Whenever the ERCA generates a new ~~European~~ root key pair, it shall create a new self-signed root certificate for the new ~~European~~ Root public key. The validity period of a ~~European~~ root certificate shall be 34 years plus 3 months.

Note: The introduction of a new root key pair also implies that ERCA will generate a new motion sensor master key and a new DSRC master key, see sections ~~9.2.2.2 and 9.2.3.2~~ **9.2.1.2 and 9.2.2.2**.

CSM_57 Before generating a new ~~European~~ root key pair, the ERCA shall conduct an analysis of the cryptographic strength that is needed for the new key pair, given it should stay secure for the next 34 years. If found necessary, the ERCA shall switch to a cipher suite that is stronger than the current one, as specified in ~~TCS_192~~ CSM_50.

CSM_58 Whenever it generates a new ~~European~~ root key pair, the ERCA shall create a link certificate for the new ~~European~~ Root public key and sign it with the previous ~~European~~ Root private key. The validity period of the link certificate shall be 17 years **plus 3 months**. This is shown in **Figure 1** in section **9.1.7** as well.

Note: Since a link certificate contains the ERCA generation *X* public key and is signed with the ERCA generation *X-1* private key, a link certificate offers equipment issued under generation *X-1* a method to trust equipment issued under generation *X*.

CSM_59 The ERCA shall not use the private key of a root key pair for any purpose after the moment a new root key certificate becomes valid.

CSM_60 At any moment in time, the ERCA shall dispose of the following cryptographic keys and certificates:

- The current EUR key pair and corresponding certificate
- All previous EUR certificates to be used for the verification of MSCA certificates that are still valid
- Link certificates for all generations of EUR certificates except the first one

9.1.3 ~~Member State~~Contracting Party Level

CSM_61 At ~~Member State~~ **Contracting Party** level, all ~~Member State~~ **Contracting Parties** required to sign tachograph card certificates shall generate one or more unique ECC key pairs designated as MSCA_Card. All ~~Member States~~ **Contracting Parties** required to

sign certificates for vehicle units or external GNSS facilities shall additionally generate one or more unique ECC key pairs designated as MSCA_VU-EGF.

CSM_62 The task of generating ~~Member State Contracting Party~~ key pairs shall be handled by a ~~Member State Contracting Party~~ Certificate Authority (MSCA). Whenever a MSCA generates a ~~Member State Contracting Party~~ key pair, it shall send the public key to the ERCA in order to obtain a corresponding ~~Member State Contracting Party~~ certificate signed by the ERCA.

CSM_63 An MSCA shall choose the strength of a ~~Member State Contracting Party~~ key pair equal to the strength of the ~~European~~ root key pair used to sign the corresponding ~~Member State Contracting Party~~ certificate.

CSM_64 An MSCA_VU-EGF key pair, if present, shall consist of private key MSCA_VU-EGF.SK and public key MSCA_VU-EGF.PK. An MSCA shall use the MSCA_VU-EGF.SK private key exclusively to sign the public key certificates of vehicle units and external GNSS facilities.

CSM_65 An MSCA_Card key pair shall consist of private key MSCA_Card.SK and public key MSCA_Card.PK. An MSCA shall use the MSCA_Card.SK private key exclusively to sign the public key certificates of tachograph cards.

CSM_66 An MSCA shall keep records of all signed VU certificates, external GNSS facility certificates and card certificates, together with the identification of the equipment for which each certificate is intended.

CSM_67 The validity period of an MSCA_VU-EGF certificate shall be 17 years plus 3 months. The validity period of an MSCA_Card certificate shall be 7 years plus 1 month.

CSM_68 As shown in **Figure 1** in section **9.1.7**, the private key of a MSCA_VU-EGF key pair and the private key of a MSCA_Card key pair shall have a key usage period of two years.

CSM_69 An MSCA shall not use the private key of an MSCA_VU-EGF key pair for any purpose after the moment its usage period has ended. Neither shall an MSCA use the private key of an MSCA_Card key pair for any purpose after the moment its usage period has ended.

CSM_70 At any moment in time, an MSCA shall dispose of the following cryptographic keys and certificates:

- The current MSCA_Card key pair and corresponding certificate
- All previous MSCA_Card certificates to be used for the verification of the certificates of tachograph cards that are still valid
- The current EUR certificate necessary for the verification of the current MSCA certificate
- All previous EUR certificates necessary for the verification of all MSCA certificates that are still valid

CSM_71 If an MSCA is required to sign certificates for vehicle units or external GNSS facilities, it shall additionally dispose of the following keys and certificates:

The current MSCA_VU-EGF key pair and corresponding certificate

All previous MSCA_VU-EGF public keys to be used for the verification of the certificates of VUs or external GNSS facilities that are still valid

9.1.4 Equipment Level: Vehicle Units

CSM_72 Two unique ECC key pairs shall be generated for each vehicle unit, designated as VU_MA and VU_Sign. This task is handled by VU manufacturers. Whenever a VU key pair is generated, the party generating the key shall send the public key to ~~the MSCA of the county in which it resides~~ **its MSCA**, in order to obtain a corresponding VU certificate signed by the MSCA. The private key shall be used only by the vehicle unit.

CSM_73 The VU_MA and VU_Sign certificates of a given vehicle unit shall have the same Certificate Effective Date.

CSM_74 A VU manufacturer shall choose the strength of a VU key pair equal to the strength of the MSCA key pair used to sign the corresponding VU certificate.

CSM_75 A vehicle unit shall use its VU_MA key pair, consisting of private key VU_MA.SK and public key VU_MA.PK, exclusively to perform VU Authentication towards tachograph cards and external GNSS facilities, as specified in sections **10.3 and 11.4 of this Appendix sub-appendix**.

CSM_76 A vehicle unit shall be capable of generating ephemeral ECC key pairs and shall use an ephemeral key pair exclusively to perform session key agreement with a tachograph card or external GNSS facility, as specified in sections **10.4 and 11.4 of this Appendix sub-appendix**.

CSM_77 A vehicle unit shall use the private key VU_Sign.SK of its VU_Sign key pair exclusively to sign downloaded data files, as specified in chapter **14 of this Appendix sub-appendix**. The corresponding public key VU_Sign.PK shall be used exclusively to verify signatures created by the vehicle unit.

CSM_78 As shown in **Figure 1** in section **9.1.7**, the validity period of a VU_MA certificate shall be 15 years and 3 months. The validity period of a VU_Sign certificate shall also be 15 years and 3 months.

Notes:

The extended validity period of a VU_Sign certificate allows a Vehicle Unit to create valid signatures over downloaded data during the first three months after it has expired, ~~as required in Regulation (EU)~~.

The extended validity period of a VU_MA certificate is needed to allow the VU to authenticate to a control card or a company card during the first three months after it has expired, such that it is possible to perform a data download.

CSM_79 A vehicle unit shall not use the private key of a VU key pair for any purpose after the corresponding certificate has expired.

CSM_80 The VU key pairs (except ephemeral keys pairs) and corresponding certificates of a given vehicle unit shall not be replaced or renewed in the field once the vehicle unit has been put in operation.

Notes:

Ephemeral key pairs are not included in this requirement, as a new ephemeral key pair is generated by a VU each time Chip Authentication and session key agreement is performed, see section **10.4**. Note that ephemeral key pairs do not have corresponding certificates.

This requirement does not forbid the possibility of replacing static VU key pairs during a refurbishment or repair in a secure environment controlled by the VU manufacturer.

CSM_81 When put in operation, vehicle units shall contain the following cryptographic keys and certificates:

- The VU_MA private key and corresponding certificate

- The VU_Sign private key and corresponding certificate
- The MSCA_VU-EGF certificate containing the MSCA_VU-EGF.PK public key to be used for verification of the VU_MA certificate and VU_Sign certificate
- The EUR certificate containing the EUR.PK public key to be used for verification of the MSCA_VU-EGF certificate
- The EUR certificate whose validity period directly precedes the validity period of the EUR certificate to be used to verify the MSCA_VU-EGF certificate, if existing
- The link certificate linking these two EUR certificates, if existing

CSM_82 In addition to the cryptographic keys and certificates listed in ~~TCS_223~~ **CSM_81**, vehicle units shall also contain the keys and certificates specified in Part A of this ~~Appendix~~ **sub-appendix**, allowing a vehicle unit to interact with first-generation tachograph cards.

9.1.5 Equipment Level: Tachograph Cards

CSM_83 One unique ECC key pair, designated as Card_MA, shall be generated for each tachograph card. A second unique ECC key pair, designated as Card_Sign, shall additionally be generated for each driver card and each workshop card. This task may be handled by card manufacturers or card personalisers. Whenever a card key pair is generated, the party generating the key shall send the public key to ~~the MSCA of the country in which it resides~~ **its MSCA**, in order to obtain a corresponding card certificate signed by the MSCA. The private key shall be used only by the tachograph card.

CSM_84 The Card_MA and Card_Sign certificates of a given driver card or workshop card shall have the same Certificate Effective Date.

CSM_85 A card manufacturer or card personaliser shall choose the strength of a card key pair equal to the strength of the MSCA key pair used to sign the corresponding card certificate.

CSM_86 A tachograph card shall use its Card_MA key pair, consisting of private key Card_MA.SK and public key Card_MA.PK, exclusively to perform mutual authentication and session key agreement towards vehicle units, as specified in sections **10.3 and 10.4 of this Appendix sub-appendix**.

CSM_87 A driver card or workshop card shall use the private key Card_Sign.SK of its Card_Sign key pair exclusively to sign downloaded data files, as specified in chapter **14** of this ~~Appendix~~ **sub-appendix**. The corresponding public key Card_Sign.PK shall be used exclusively to verify signatures created by the card.

CSM_88 The validity period of a Card_MA certificate shall be as follows:

- For driver cards: 5 years
- For company cards: \geq 5 years
- For control cards: 2 years
- For workshop cards: 1 year

CSM_89 The validity period of a Card_Sign certificate shall be as follows:

- For driver cards: 5 years and 1 month
- For workshop cards: 1 year and 1 month

Note: the extended validity period of a Card_Sign certificate allows a driver card to create valid signatures over downloaded data during the first month after it has expired. ~~This is~~

~~necessary in view of Regulation (EU) No 58/2010, which requires that a data download from a driver card must be possible up to 28 days after the last data has been recorded.~~

CSM_90 The key pairs and corresponding certificates of a given tachograph card shall not be replaced or renewed once the card has been issued.

CSM_91 When issued, tachograph cards shall contain the following cryptographic keys and certificates:

- The Card_MA private key and corresponding certificate
- For driver cards and workshop cards additionally: the Card_Sign private key and corresponding certificate
- The MSCA_Card certificate containing the MSCA_Card.PK public key to be used for verification of the Card_MA certificate and Card_Sign certificate
- The EUR certificate containing the EUR.PK public key to be used for verification of the MSCA_Card certificate.
- The EUR certificate whose validity period directly precedes the validity period of the EUR certificate to be used to verify the MSCA_Card certificate, if existing.
- The link certificate linking these two EUR certificates, if existing.
- **Additionally, for control cards, company cards and workshop cards only, and only if such cards are issued during the first three months of the validity period of a new EUR certificate: the EUR certificate that is two generations older, if existing.**

Note to last bullet: For example, in the first three months of the ERCA(3) certificate (see Figure 1), the mentioned cards shall contain the ERCA(1) certificate. This is needed to ensure that these cards can be used to perform data downloads from ERCA(1) VUs whose normal 15-year life period plus the 3-months data downloading period expires during these months; see the last bullet of requirement 13) in Appendix 1C.

CSM_92 In addition to the cryptographic keys and certificates listed in ~~TCS_233~~ **CSM_91**, tachograph cards shall also contain the keys and certificates specified in Part A of this ~~Appendix~~ **sub-appendix**, allowing these cards to interact with first-generation VUs.

9.1.6 Equipment Level: External GNSS Facilities

CSM_93 One unique ECC key pair shall be generated for each external GNSS facility, designated as EGF_MA. This task is handled by external GNSS facility manufacturers. Whenever an EGF_MA key pair is generated, **the party generating the key shall be sent the public key shall be sent to the its MSCA of the country in which it resides** in order to obtain a corresponding EGF_MA certificate signed by the MSCA. The private key shall be used only by the external GNSS facility.

CSM_94 An EGF manufacturer shall choose the strength of an EGF_MA key pair equal to the strength of the MSCA key pair used to sign the corresponding EGF_MA certificate.

CSM_95 An external GNSS facility shall use its EGF_MA key pair, consisting of private key EGF_MA.SK and public key EGF_MA.PK, exclusively to perform mutual authentication and session key agreement towards vehicle units, as specified in section **11.4 of this Appendix sub-appendix**.

CSM_96 The validity period of an EGF_MA certificate shall be 15 years.

CSM_97 An external GNSS facility shall not use the private key of its EGF_MA key pair for coupling to a vehicle unit after the corresponding certificate has expired.

Note: as explained in section **11.3.3**, an EGF may potentially use its private key for mutual authentication towards the VU it is already coupled to, even after the corresponding certificate has expired.

CSM_98 The EGF_MA key pair and corresponding certificate of a given external GNSS facility shall not be replaced or renewed in the field once the EGF has been put in operation.

Note: This requirement does not forbid the possibility of replacing EGF key pairs during a refurbishment or repair in a secure environment controlled by the EGF manufacturer.

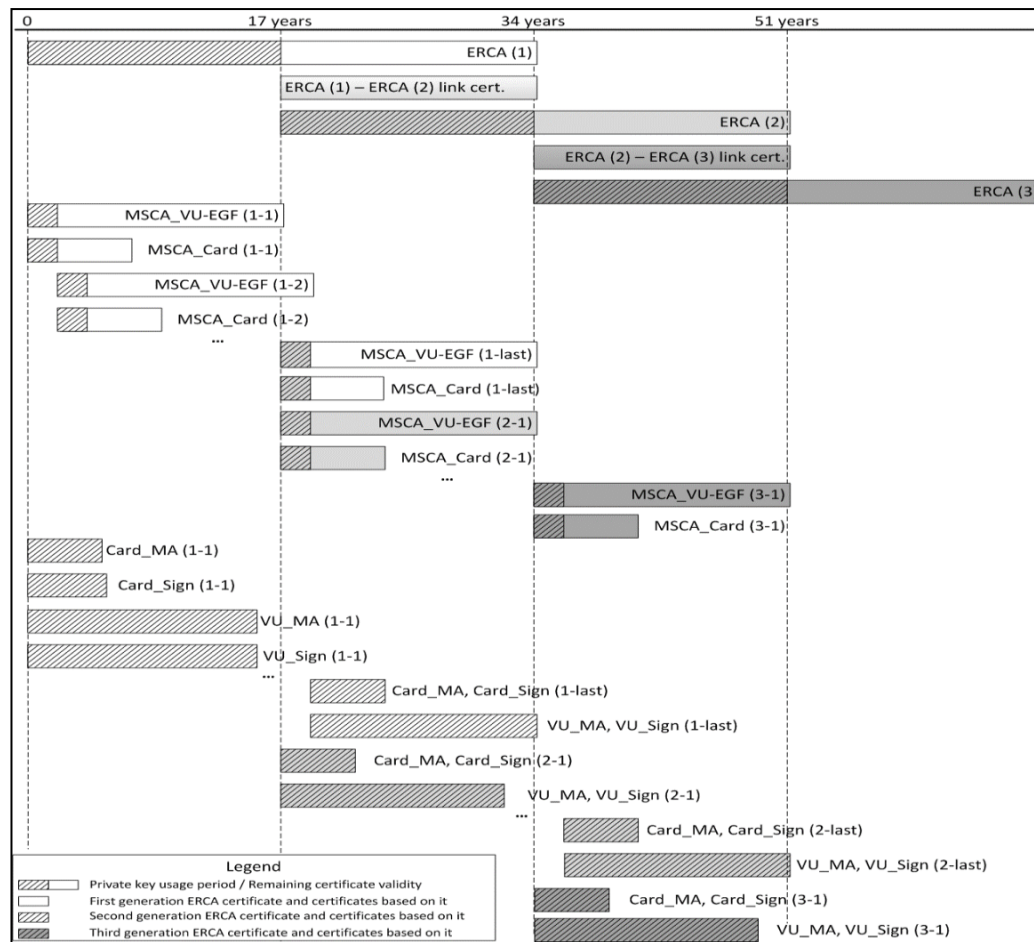
CSM_99 When put in operation, an external GNSS facility shall contain the following cryptographic keys and certificates:

- The EGF_MA private key and corresponding certificate
- The MSCA_VU-EGF certificate containing the MSCA_VU-EGF.PK public key to be used for verification of the EGF_MA certificate
- The EUR certificate containing the EUR.PK public key to be used for verification of the MSCA_VU-EGF certificate
- The EUR certificate whose validity period directly precedes the validity period of the EUR certificate to be used to verify the MSCA_VU-EGF certificate, if existing
- The link certificate linking these two EUR certificates, if existing

9.1.7 Overview: Certificate Replacement

Figure 1 below shows how different generations of ERCA root certificates, ERCA link certificates, MSCA certificates and equipment (VU and card) certificates are issued and used over time:

Figure 1
Issuance and usage of different generations of ERCA root certificates, ERCA link certificates, MSCA certificates and equipment certificates



Notes to Figure 1:

1. Different generations of the root certificate are indicated by a number in brackets. E.g. ERCA (1) is the first generation of ERCA root certificate; ERCA (2) is the second generation, etc.
2. Other certificates are indicated by two numbers in brackets, the first one indicating the root certificate generation under which they are issued, the second one the generation of the certificate itself. E.g. MSCA_Card (1-1) is the first MSCA_Card certificate issued under ERCA (1); MSCA_Card (2-1) is the first MSCA_Card certificate issued under ERCA (2); MSCA_Card (2-last) is the last MSCA_Card certificate issued under ERCA (2); Card_MA(2-1) is the first Card certificate for mutual authentication that is issued under ERCA (2), etc.
3. The MSCA_Card (2-1) and MSCA_Card (1-last) certificates are issued at almost but not exactly the same date. MSCA_Card (2-1) is the first MSCA_Card certificate issued under ERCA (2) and will be issued slightly later than MSCA_Card (1-last), the last MSCA_Card certificate under ERCA (1).
4. As shown in the figure, the first VU and Card certificates issued under ERCA (2) will appear almost two years before the last VU and Card certificates issued under ERCA (1) will appear. This is because of the fact that VU and Card certificates are issued under an MSCA

certificate, not directly under the ERCA certificate. The MSCA (2-1) certificate will be issued directly after ERCA (2) becomes valid, but the MSCA (1-last) certificate will be issued only slightly before that time, at the last moment the ERCA (1) certificate is still valid. Therefore, these two MSCA certificates will have almost the same validity period, despite the fact that they are of different generations.

5. The validity period shown for cards is the one for driver cards (5 years).
6. To save space, the difference in validity period between the Card_MA and Card_Sign certificates and ~~between the VU_MA_ and VU_Sign certificates~~ is shown only for the first generation.

9.2. Symmetric Keys

9.2.1 Keys for Securing VU – Motion Sensor Communication

~~9.2.2 Keys for Securing DSRC Communication~~

9.2.1.1 General

Note: readers of this section are supposed to be familiar with the contents of [ISO 16844-3] describing the interface between a vehicle unit and a motion sensor. The pairing process between a VU and a motion sensor is described in detail in chapter 12 of this ~~Appendix~~ **sub-appendix**.

CSM_100 A number of symmetric keys is needed for pairing vehicle units and motion sensors, for mutual authentication between vehicle units and motion sensors and for encrypting communication between vehicle units and motion sensors, as shown in ~~Table 45~~ Table 3. All of these keys shall be AES keys, with a key length equal to the length of the motion sensor master key, which shall be linked to the length of the (foreseen) ~~European~~ **Root** key pair as described in ~~TCS_192~~ **CSM_50**.

Table 45-3
Keys for securing vehicle unit - motion sensor communication

Key	Symbol	Generated by	Generation method	Stored by
Motion Sensor Master Key – VU part	K _{M-VU}	ERCA	Random	ERCA, MSCAs involved in issuing VUs certificates, VU manufacturers, vehicle units
Motion Sensor Master Key – Workshop part	K _{M-WC}	ERCA	Random	ERCA, MSCAs, card manufacturers, workshop cards
Motion Sensor Master Key	K _M	Not independently generated	Calculated as $K_M = K_{M-VU} \text{ XOR } K_{M-WC}$	ERCA, MSCAs involved in issuing motion sensors keys (optionally)*
Identification Key	K _{ID}	Not independently generated	Calculated as $K_{ID} = K_M \text{ XOR } CV$, where CV is specified in TCS_248 CSM_106	ERCA, MSCAs involved in issuing motion sensors keys (optionally)*
Pairing Key	K _P	Motion sensor manufacturer	Random	One motion sensor
Session Key	K _S	VU (during pairing of VU)	Random	One VU and one motion sensor

Key	Symbol	Generated by	Generation method	Stored by
		and motion sensor)		

*Storage of K_M and K_{ID} is optional, as these keys can be derived from K_{M-VU} , K_{M-WC} and CV .

CSM_101 The ~~European~~ Root Certificate Authority shall generate K_{M-VU} and K_{M-WC} , two random and unique AES keys from which the motion sensor master key K_M can be calculated as $K_{M-VU} \text{ XOR } K_{M-WC}$. The ERCA shall communicate K_M , K_{M-VU} and K_{M-WC} to ~~Member State~~ **Contracting Party** Certificate Authorities upon their request.

CSM_102 The ERCA shall assign to each motion sensor master key K_M a unique version number, which shall also be applicable for the constituting keys K_{M-VU} and K_{M-WC} and for the related identification key K_{ID} . The ERCA shall inform the MSCAs about the version number when sending K_{M-VU} and K_{M-WC} to them.

Note: The version number is used to distinguish different generations of these keys, as explained in detail in section ~~9.2.2.2~~ **9.2.1.2**.

CSM_103 A ~~Member State~~ **Contracting Party** Certificate Authority shall forward K_{M-VU} , together with its version number, to vehicle unit manufacturers upon their request. The VU manufacturers shall insert K_{M-VU} and its version number in all manufactured VUs.

CSM_104 A ~~Member State~~ **Contracting Party** Certificate Authority shall ensure that K_{M-WC} , together with its version number, is inserted in every workshop card issued under its responsibility.

Notes:

- See the description of data type SensorInstallationSecData_a in ~~Member State~~ **Sub-appendix 2**.
- as explained in section ~~9.2.2.2~~ **9.2.1.2**, in fact multiple generations of K_{M-WC} may have to be inserted in a single workshop card.

CSM_105 In addition to the AES key specified in ~~TCS_246~~ **CSM_104**, a MSCA shall ensure that the TDES key K_{mwc} , specified in requirement CSM_037 in Part A of this ~~Appendix~~ **sub-appendix**, is inserted in every workshop card issued under its responsibility.

Notes:

- This allows a second-generation workshop card to be used for coupling a first-generation VU.
- A second-generation workshop card will contain two different applications, one complying with Part B of this ~~Appendix~~ **sub-appendix** and one complying with Part A. The latter will contain the TDES key K_{mwc} .

CSM_106 An MSCA involved in issuing motion sensors shall derive the identification key from the motion sensor master key by XORing it with a constant vector CV . The value of CV shall be as follows:

For 128-bit motion sensor master keys: $CV = \text{'B6 44 2C 45 0E F8 D3 62 0B 7A 8A 97 91 E4 5ED 83'}$

For 192-bit motion sensor master keys: $CV = \text{'72 AD EA FA 00 BB F4 EE F4 99 15 70 5B 7E EE BB 1C 54 ED 46 8B 0E F8 25'}$

For 256-bit motion sensor master keys: $CV = \text{'1D 74 DB F0 34 C7 37 2F 65 55 DE D5 DC D1 9A C3 23 D6 A6 25 64 CD BE 2D 42 0D 85 D2 32 63 AD 60'}$

Note: the constant vectors have been generated as follows:

Pi_10 = first 10 bytes of the decimal portion of the mathematical constant π = '24 3F 6A 88 85 A3 08 D3 13 19'

CV_128-bits = first 16 bytes of SHA-256(Pi_10)

CV_192-bits = first 24 bytes of SHA-384(Pi_10)

CV_256-bits = first 32 bytes of SHA-512(Pi_10)

CSM_107 Each motion sensor manufacturer shall generate a random and unique pairing key K_P for every motion sensor, and shall send each pairing key to ~~its a Member State Contracting Party~~ Certificate Authority. The MSCA shall encrypt each pairing key separately with the motion sensor master key K_M and shall return the encrypted key to the motion sensor manufacturer. For each encrypted key, the MSCA shall notify the motion sensor manufacturer of the version number of the associated K_M .

Note: as explained in section ~~9.2.2.2~~ **9.2.1.2**, in fact a motion sensor manufacturer may have to generate multiple unique pairing keys for a single motion sensor.

CSM_108 Each motion sensor manufacturer shall generate a unique serial number for every motion sensor, and shall send all serial numbers to ~~its a Member State Contracting Party~~ Certificate Authority. The MSCA shall encrypt each serial number separately with the identification key K_{ID} and shall return the encrypted serial number to the motion sensor manufacturer. For each encrypted serial number, the MSCA shall notify the motion sensor manufacturer of the version number of the associated K_{ID} .

CSM_109 For requirements ~~TCS_249~~ **CSM_107** and ~~TCS_250~~ **CSM_108**, the MSCA shall use the AES algorithm in the Cipher Block Chaining mode of operation, as defined in [ISO 10116], with an interleave parameter $m = 1$ and an initialization vector $SV = '00' \{16\}$, i.e. sixteen bytes with binary value 0. When necessary, the MSCA shall use padding method 2 defined in [ISO 9797-1].

CSM_110 The motion sensor manufacturer shall store the encrypted pairing key and the encrypted serial number in the intended motion sensor, together with the corresponding plain text values and the version number of K_M and K_{ID} used for encrypting.

Note: as explained in section ~~9.2.2.2~~ **9.2.1.2**, in fact a motion sensor manufacturer may have to insert multiple encrypted pairing keys and multiple encrypted serial numbers in a single motion sensor.

CSM_111 In addition to the AES-based cryptographic material specified in ~~TCS_252~~ **CSM_110**, a motion sensor manufacturer may also store in each motion sensor the TDES-based cryptographic material specified in requirement CSM_037 in Part A of this ~~Appendix~~ **sub-appendix**.

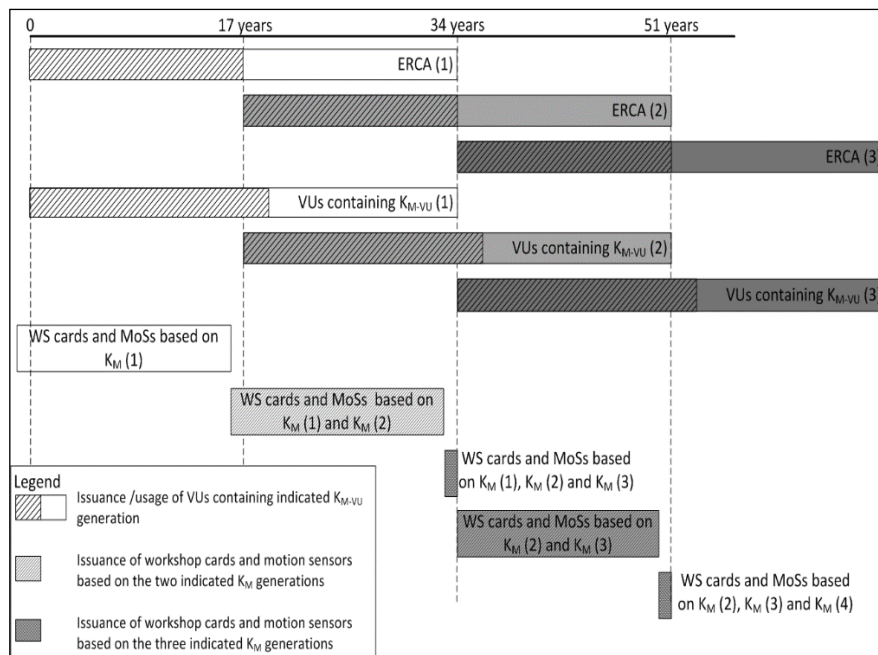
Note: doing so will allow a second-generation motion sensor to be coupled to a first-generation VU.

CSM_112 The length of the session key K_S generated by a VU during the pairing to a motion sensor shall be linked to the length of its K_{M-VU} , as described in ~~TCS_492~~ **CSM_50**.

9.2.1.2 Motion Sensor Master Key Replacement in Second-Generation Equipment

CSM_113 Each motion sensor master key and all related keys (see ~~Table 45~~ **Table 3**) is associated to a particular generation of the ERCA root key pair. These keys shall therefore be replaced every 17 years. The validity period of each motion sensor master key generation shall begin one year before the associated ERCA root key pair becomes valid and shall end when the associated ERCA root key pair expires. This is depicted in **Figure 2**.

Figure 1
Issuance and usage of different generations of the motion sensor master key in vehicle units, motions sensors and workshop cards



CSM_114 At least one year before generating a new ~~European~~ root key pair, as described in ~~TCS_198~~ CSM_56, the ERCA shall generate a new motion sensor master key K_M by generating a new K_{M-VU} and K_{M-WC} . The length of the motion sensor master key shall be linked to the foreseen strength of the new ~~European~~ root key pair, according to ~~TCS_192~~ CSM_50. The ERCA shall communicate the new K_M , K_{M-VU} and K_{M-WC} to the MSCAs upon their request, together with their version number.

CSM_115 An MSCA shall ensure that all valid generations of K_{M-WC} are stored in every workshop card issued under its authority, together with their version numbers, as shown in Figure 2.

Note: this implies that in the last year of the validity period of an ERCA certificate, workshop cards will be issued with three different generations of K_{M-WC} , as shown in **Figure 2**.

CSM_116 In relation to the process described in ~~TCS_249~~ CSM_107 and ~~TCS_250~~ CSM_108 above: an MSCA shall encrypt each pairing key K_P it receives from a motion sensor manufacturer separately with each valid generation of the motion sensor master key K_M . An MSCA shall also encrypt each serial number it receives from a motion sensor manufacturer separately with each valid generation of the identification key K_{ID} . A motion sensor manufacturer shall store all encryptions of the pairing key and all encryptions of the serial number in the intended motion sensor, together with the corresponding plain text values and the version number(s) of K_M and K_{ID} used for encrypting.

Note: This implies that in the last year of the validity period of an ERCA certificate, motion sensors will be issued with encrypted data based on three different generations of K_M , as shown in Figure 2.

CSM_117 In relation to the process described in ~~TCS_249~~ CSM_107 above: since the length of the pairing key K_P shall be linked to the length of K_M (see ~~TCS_242~~ CSM_100), a motion sensor manufacturer may have to generate up to three different pairing keys (of different lengths) for one motion sensor, in case subsequent generations of K_M have different

lengths. In such a case, the manufacturer shall send each pairing key to the MSCA. The MSCA shall ensure that each pairing key is encrypted with the correct generation of the motion sensor master key, i.e. the one having the same length.

Note: In case the motion sensor manufacturer chooses to generate a TDES-based pairing key for a second-generation motion sensor (see ~~TCS_253~~ CSM_111), the manufacturer shall indicate to the MSCA that the TDES-based motion sensor master key must be used for encrypting this pairing key. This is because the length of a TDES key may be equal to that of an AES key, so the MSCA cannot judge from the key length alone.

CSM_118 Vehicle unit manufacturers shall insert only one generation of K_{M-VU} in each vehicle unit, together with its version number. This K_{M-VU} generation shall be linked to the ERCA certificate upon which the VU's certificates are based.

Notes:

- A vehicle unit based on the generation X ERCA certificate shall only contain the generation X K_{M-VU} , even if it is issued after the start of the validity period of the generation $X+1$ ERCA certificate. This is shown in Figure 2.
- A VU of generation X cannot be paired to a motion sensor of generation $X-1$.
- Since workshop cards have a validity period of one year, the result of ~~TCS_255~~ ~~TCS_260~~ CSM_113 – CSM_118 is that all workshop cards will contain the new K_{M-WC} at the moment the first VU containing the new K_{M-VU} is issued. Therefore, such a VU will always be able to calculate the new K_M . Moreover, by that time most new motion sensors will contain encrypted data based on the new K_M as well.

9.2.2 Keys for Securing DSRC Communication

9.2.2.1 General

CSM_119 The authenticity and confidentiality of data communicated from a vehicle unit to a control authority over a DSRC remote communication channel shall be ensured by means of a set of VU-specific AES keys derived from a single DSRC master key, K_{M-DSRC} .

CSM_120 The DSRC master key K_{M-DSRC} shall be an AES key that is securely generated, stored and distributed by the ERCA. The key length may be 128, 192 or 256 bits and shall be linked to the length of the ~~European~~ root key pair, as described in ~~TCS_192~~ CSM_50.

CSM_121 The ERCA shall communicate the DSRC master key to ~~Member State Contracting Party~~ Certificate Authorities upon their request in a secure manner, to allow them to derive VU-specific DSRC keys and to ensure that the DSRC master key is inserted in all control cards and workshop cards issued under their responsibility.

CSM_112 The ERCA shall assign to each DSRC master key a unique version number. The ERCA shall inform the MSCAs about the version number when sending the DSRC master key to them.

Note: The version number is used to distinguish different generations of the DSRC master key, as explained in detail in section ~~9.2.3.2~~ 9.2.2.2.

CSM_123 For every vehicle unit, the vehicle unit manufacturer shall create a unique VU serial number and shall send this number to its ~~Member State Contracting Party~~ Certificate Authority in a request to obtain a set of two VU-specific DSRC keys. The VU serial number shall have data type `VuSerialNumber`, and the ~~Distinguished Encoding Rule (DER) according to (ISO 88251)~~ shall be used for encoding.

Note:

- **This VU serial number shall be identical to the `vuSerialNumber` element of `VuIdentification`, see Sub-appendix 1 and to the Certificate Holder Reference in the VU's certificates.**
- **The VU serial number may not be known at the moment a vehicle unit manufacturer requests the VU-specific DSRC keys. In this case, the VU manufacturer shall send instead the unique certificate request ID it used when requesting the VU's certificates; see CSM_153. This certificate request ID shall therefore be equal to the Certificate Holder Reference in the VU's certificates.**

CSM_124 Upon receiving a request for VU-specific DSRC keys, the MSCA shall derive two AES keys for the vehicle unit, called `K_VUDSRC_ENC` and `K_VUDSRC_MAC`. These VU-specific keys shall have the same length as the DSRC master key. The MSCA shall use the key derivation function defined in [RFC 5869]. The hash function that is necessary to instantiate the HMAC-Hash function shall be linked to the length of the DSRC master key, as described in 0. The key derivation function in [RFC 5869] shall be used as follows:

Step 1 (Extract):

- $PRK = \text{HMAC-Hash}(salt, IKM)$ where *salt* is an empty string ‘’ and *IKM* is KM_{DSRC} .

Step 2 (Expand):

- $OKM = T(1)$, where
 $T(1) = \text{HMAC-Hash}(PRK, T(0) \parallel info \parallel '01')$ with
 $T(0) =$ an empty string (‘’)

info = VU serial number or certificate request ID, as specified in ~~TCS_265~~ CSM_123

`K_VUDSRC_ENC` = first *L* octets of *OKM* and

`K_VUDSRC_MAC` = last *L* octets of *OKM*

where *L* is the required length of `K_VUDSRC_ENC` and `K_VUDSRC_MAC` in octets.

CSM_125 The MSCA shall distribute `K_VUDSRC_ENC` and `K_VUDSRC_MAC` to the VU manufacturer in a secure manner for insertion in the intended vehicle unit.

CSM_126 When issued, a vehicle unit shall have stored `K_VUDSRC_ENC` and `K_VUDSRC_MAC` in its secure memory, in order to be able to ensure the integrity, authenticity and confidentiality of data sent over the remote communication channel. A vehicle unit shall also store the version number of the DSRC master key used to derive these VU-specific keys.

CSM_127 When issued, control cards and workshop cards shall have stored KM_{DSRC} in their secure memory, in order to be able to verify the integrity and authenticity of data sent by a VU over the remote communication channel and to decrypt this data. Control cards and workshop cards shall also store the version number of the DSRC master key.

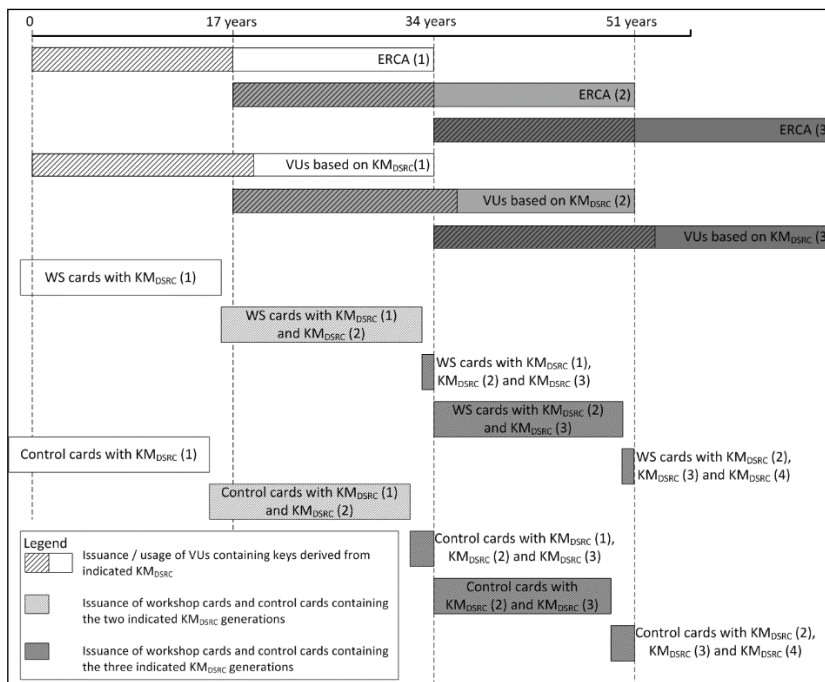
Note: as explained in section ~~9.2.3.2~~ 9.2.2.2, in fact multiple generations of KM_{DSRC} may have to be inserted in a single workshop card or control card.

CSM_128 The MSCA shall keep records of all VU-specific DSRC keys it generated, their version number and the ~~identification of the VU for which each set of keys is intended~~ **VU serial number or certificate request ID used in deriving them.**

9.2.2.2 ~~9.2.3.2~~ DSRC Master Key Replacement

CSM_129 Each DSRC master key is associated to a particular generation of the ERCA root key pair. The ERCA shall therefore replace the DSRC master key every 17 years. The validity period of each DSRC master key generation shall begin two years before the associated ERCA root key pair becomes valid and shall end when the associated ERCA root key pair expires. This is depicted in Figure 3.

Figure 3
Issuance and usage of different generations of the DSRC master key in vehicle units, workshop cards and control cards



CSM_130 At least two years before generating a new ~~European~~ root key pair, as described in ~~TCS_198~~ CSM_56, the ERCA shall generate a new DSRC master key. The length of the DSRC key shall be linked to the foreseen strength of the new ~~European~~ root key pair, according to ~~TCS_192~~ CSM_50. The ERCA shall communicate the new DSRC master key to the MSCAs upon their request, together with its version number.

CSM_131 An MSCA shall ensure that all valid generations of KM_DSRC are stored in every control card issued under its authority, together with their version numbers, as shown in **Figure 3**.

Note: this implies that in the last two years of the validity period of an ERCA certificate, control cards will be issued with three different generations of KM_DSRC, as shown in **Figure 3**.

CSM_132 An MSCA shall ensure that all generations of KM_DSRC that have been valid for at least a year and are still valid, are stored in every workshop card issued under its authority, together with their version numbers, as shown in **Figure 3**.

Note: this implies that in the last year of the validity period of an ERCA certificate, workshop cards will be issued with three different generations of KM_DSRC, as shown in **Figure 3**.

CSM_133 Vehicle unit manufacturers shall insert only one set of VU-specific DSRC keys into each vehicle unit, together with its version number. This set of keys shall be derived from the KM_DSRC generation linked to the ERCA certificate upon which the VU's certificates are based.

Notes:

This implies that a vehicle unit based on the generation *X* ERCA certificate shall only contain the generation *X* K_VU_DSRC_ENC and K_VU_DSRC_MAC, even if the VU is issued after the start of the validity period of the generation *X+1* ERCA certificate. This is shown in **Figure 3**.

Since workshop cards have a validity period of one year and control cards of two years, the result of ~~TCS_273 – TCS_275~~ **CSM_131 – CSM_133** is that all workshop cards and control cards will contain the new DSRC master key at the moment the first VU containing VU-specific keys based on that master key will be issued.

9.3. Certificates

9.3.1 General

CSM_134 All certificates in the ~~European~~ Smart Tachograph system shall be self-descriptive, card-verifiable (CV) certificates according to [ISO 7816-4] and [ISO 7816-8].

CSM_135 The Distinguished Encoding Rules (DER) according to [ISO 8825-1] shall be used to encode ~~both ASN.1 data structures and (application specific)~~ the data objects within certificates. **Table 4 shows the full certificate encoding, including all tag and length bytes.**

Note: this encoding results in a Tag-Length-Value (TLV) structure as follows:

- Tag: The tag is encoded in one or two octets and indicates the content.
- Length: The length is encoded as an unsigned integer in one, two, or three octets, resulting in a maximum length of 65535 octets. The minimum number of octets shall be used.
- Value: The value is encoded in zero or more octets

9.3.2 Certificate Content

CSM_136 All certificates shall have the structure shown in the certificate profile in **Table 4 46**.

Table 4
Certificate Profile version 1

<i>Field</i>	<i>Field ID</i>	<i>Tag</i>	<i>Length (bytes)</i>	<i>ASN.1 data type (see Appendix 1)</i>
ECC Certificate	C	'7F 21'	var	
ECC Certificate Body	B	'7F 4E'	var	
Certificate Profile Identifier	CPI	'5F 29'	'01'	INTEGER(0..255)
Certificate Authority Reference	CAR	'42'	'08'	KeyIdentifier
Certificate Holder Authorisation	CHA	'5F 4C'	'07'	CertificateHolderAuthorisation
Public Key	PK	'7F 49'	var	
Domain Parameters	DP	'06'	var	OBJECT IDENTIFIER
Public Point	PP	'86'	var	OCTET STRING
Certificate Holder Reference	CHR	'5F 20'	'08'	KeyIdentifier
Certificate Effective Date	CEfD	'5F 25'	'04'	TimeReal
Certificate Expiration Date	CExD	'5F 24'	'04'	TimeReal
ECC Certificate Signature	S	'5F 37'	var	OCTET STRING

Note: the Field ID will be used in later sections of this Appendix **sub-appendix** to indicate individual fields of a certificate, e.g. X.CAR is the Certificate Authority Reference mentioned in the certificate of user X.

9.3.2.1 Certificate Profile Identifier

CSM_137 Certificates shall use a Certificate Profile Identifier to indicate the certificate profile used. Version 1, as specified in **Table 4 46**, shall be identified by a value of '00'.

9.3.2.2 ~~9.3.2.1~~ Certificate Authority Reference

CSM_138 The Certificate Authority Reference shall be used to identify the public key to be used to verify the certificate signature. The Certificate Authority Reference shall therefore be equal to the Certificate Holder Reference in the certificate of the corresponding certificate authority.

CSM_139 An ERCA root certificate shall be self-signed, i.e., the Certificate Authority Reference and the Certificate Holder Reference in the certificate shall be equal.

CSM_140 For an ERCA link certificate, the Certificate Holder Reference shall be equal to the CHR of the new ERCA root certificate. The Certificate Authority Reference for a link certificate shall be equal to the CHR of the previous ERCA root certificate.

9.3.2.3 Certificate Holder Authorisation

CSM_141 The Certificate Holder Authorisation shall be used to identify the type of certificate. It consists of the six most significant bytes of the Tachograph Application ID, concatenated with the ~~type of equipment for which the certificate is intended~~ **equipment type, which indicates the type of equipment for which the certificate is intended. In the case of a VU certificate, a driver card certificate or a workshop card certificate, the equipment type is also used to differentiate between a certificate for Mutual Authentication and a certificate for creating digital signatures (see section 9.1 and Sub-appendix 1, data type EquipmentType).**

9.3.2.4 Public Key

The Public Key nests two data elements: the standardized domain parameters to be used with the public key in the certificate and the value of the public point.

CSM_142 The data element Domain Parameters shall contain one of the object identifiers specified in **Table 1 43** to reference a set of standardized domain parameters.

CSM_143 The data element Public Point shall contain the public point. Elliptic curve public points shall be converted to octet strings as specified in [TR-03111]. The uncompressed encoding format shall be used. When recovering an elliptic curve point from its encoded format, the validations described in [TR-03111] shall always be carried out.

9.3.2.5 Certificate Holder Reference

CSM_144 The Certificate Holder Reference is an identifier for the public key provided in the certificate. It shall be used to reference this public key in other certificates.

CSM_145 For card certificates and external GNSS facility certificates, the Certificate Holder Reference shall have the `ExtendedSerialNumber` data type specified in ~~Appendix~~ **Sub-appendix 1**.

CSM_146 For vehicle units, the manufacturer, when requesting a certificate, may or may not know the manufacturer-specific serial number of the VU for which that certificate and the associated private key is intended. In the first case, the Certificate Holder Reference shall

have the `ExtendedSerialNumber` data type specified in ~~Appendix~~ **Sub-appendix 1**. In the latter case, the Certificate Holder Reference shall have the `CertificateRequestID` data type specified in ~~Appendix~~ **Sub-appendix 1**.

Note: For a card certificate, the value of the CHR shall be equal to the value of the `cardExtendedSerialNumber` in `EF_ICC`; see **Sub-appendix 2**. For an EGF certificate, the value of the CHR shall be equal to the value of the `sensorGNSSSerialNumber` in `EF_ICC`; see **Sub-appendix 14**. For a VU certificate, the value of the CHR shall be equal to the `vuSerialNumber` element of `VuIdentification`, see **Sub-appendix 1**, unless the manufacturer does not know the manufacturer-specific serial number at the time the certificate is requested.

CSM_147 For ERCA and MSCA certificates, the Certificate Holder Reference shall have the `CertificationAuthorityKID` data type specified in ~~Appendix~~ **Sub-appendix 1**.

9.3.2.6 Certificate Effective Date

CSM_148 The Certificate Effective Date shall indicate the starting date and time of the validity period of the certificate. ~~The Certificate Effective Date shall be the date of the certificate generation.~~

9.3.2.7 Certificate Expiration Date

CSM_149 The Certificate Expiration Date shall indicate the end date and time of the validity period of the certificate.

9.3.2.8 Certificate Signature

CSM_150 The signature on the certificate shall be created over the encoded certificate body, including the certificate body tag and length. The signature algorithm shall be ECDSA, as specified in [DSS], using the hashing algorithm linked to the key size of the signing authority, as specified in ~~TCS_192~~ **CSM_50**. The signature format shall be plain, as specified in [TR-03111].

9.3.3 Requesting Certificates

CSM_151 When requesting a certificate, ~~a requester~~ **an MSCA** shall send the following data to ~~its Certificate Authority~~ **the ERCA**:

The Certificate Profile Identifier of the requested certificate

The Certificate Authority Reference expected to be used for signing the certificate.

The Public Key to be signed

CSM_152 In addition to the data in ~~TCS_293~~ **CSM_151**, an MSCA shall send the following data in a certificate request to the ERCA, allowing the ERCA to create the Certificate Holder Reference of the new MSCA certificate:

The numerical nation code of the Certification Authority (data type `NationNumeric` defined in ~~Appendix~~ **Sub-appendix 1**)

The alphanumerical nation code of the Certification Authority (data type `NationAlpha` defined in ~~Appendix~~ **Sub-appendix 1**)

The 1-byte serial number to distinguish the different keys of the Certification Authority in the case keys are changed

The two-byte field containing Certification Authority specific additional info

CSM_153 ~~In addition to the data in TCS_293, an~~ **An** equipment manufacturer shall send the following data in a certificate request to an MSCA, allowing the MSCA to create the Certificate Holder Reference of the new equipment certificate:

~~A manufacturer specific identifier of the equipment type~~

If known (see ~~TCS_296~~ **CSM_154**), a serial number for the equipment, unique for the manufacturer, the equipment's type and the month of manufacturing. Otherwise, a unique certificate request identifier.

The month and the year of equipment manufacturing or of the certificate request.

The manufacturer shall ensure that this data is correct and that the certificate returned by the MSCA is inserted in the intended equipment.

CSM_154 In the case of a VU, the manufacturer, when requesting a certificate, may or may not know the manufacturer-specific serial number of the VU for which that certificate and the associated private key is intended. If known, the VU manufacturer shall send the serial number to the MSCA. If not known, the manufacturer shall uniquely identify each certificate request and send this certificate request serial number to the MSCA. The resulting certificate will then contain the certificate request serial number. After inserting the certificate in a specific VU, the manufacturer shall communicate the connection between the certificate request serial number and the VU identification to the MSCA.

10. VU- Card Mutual Authentication and Secure Messaging

10.1. General

CSM_155 On a high level, secure communication between a vehicle unit and a tachograph card shall be based on the following steps:

- First, each party shall demonstrate to the other that it owns a valid public key certificate, signed by a ~~Member State~~ **Contracting Party** Certificate Authority. In turn, the MSCA public key certificate must be signed by the ~~European~~ **Root** certificate authority. This step is called certificate chain verification and is specified in detail in section **10.2**
- Second, the vehicle unit shall demonstrate to the card that it is in possession of the private key corresponding to the public key in the presented certificate. It does so by signing a random number sent by the card. The card verifies the signature over the random number. If this verification is successful, the VU is authenticated. This step is called VU Authentication and is specified in detail in section **10.3**.
- Third, both parties independently calculate two AES session keys using an asymmetric key agreement algorithm. Using one of these session keys, the card creates a message authentication code (MAC) over some data sent by the VU. The VU verifies the MAC. If this verification is successful, the card is authenticated. This step is called Card Authentication and is specified in detail in section **10.4**.
- Fourth, the VU and the card shall use the agreed session keys to ensure the confidentiality, integrity and authenticity of all exchanged messages. This is called Secure Messaging and is specified in detail in section **10.5**.

CSM_156 The mechanism described in ~~TCS_297~~ **CSM_155** shall be triggered by the vehicle unit whenever a card is inserted into one of its card slots.

10.2. Mutual Certificate Chain Verification

10.2.1 Card Certificate Chain Verification by VU

CSM_157 Vehicle units shall use the protocol depicted in Figure 4 for verifying a tachograph card's certificate chain. **For every certificate it reads from the card, the VU shall verify that the Certificate Holder Authorisation (CHA) field is correct:**

- **The CHA field of the Card certificate shall indicate a card certificate for mutual authentication (see Sub-appendix 1, data type EquipmentType).**
- **The CHA of the Card.CA certificate shall indicate an MSCA.**
- **The CHA of the Card.Link certificate shall indicate the ERCA.**

Notes to **Figure 4**:

- The Card certificates and public keys mentioned in the figure are those for mutual authentication. Section 9.1.5 denotes these as Card_MA.
- The Card.CA certificates and public keys mentioned in the figure are those for signing card certificates and it is indicated in the CAR of the Card certificate. Section 9.1.3 denotes these as MSCA_Card.
- The Card.CA.EUR certificate mentioned in the figure is the ~~European~~-root certificate that is indicated in the CAR of the Card.CA certificate.
- The Card.Link certificate mentioned in the figure is the card's link certificate, if present. As specified in section 9.1.2, this is a link certificate for a new European root key pair created by the ERCA and signed by the previous ~~European root~~ private key.
- The Card.Link.EUR certificate is the ~~European~~-root certificate that is indicated in the CAR of the Card.Link certificate.

CSM_158 As depicted in **Figure 4**, verification of the card's certificate chain shall begin upon card insertion. The vehicle unit shall read the card holder reference (cardExtendedSerialNumber) from EF ICC. The VU shall check if it knows the card, i.e., if it has successfully verified the card's certificate chain in the past and stored it for future reference. If it does, and the card certificate is still valid, the process continues with the verification of the VU certificate chain. Otherwise, the VU shall successively read from the card the MSCA_Card certificate to be used for verifying the card certificate, the Card.CA.EUR certificate to be used for verifying the MSCA_Card certificate, and possibly the link certificate, until it finds a certificate it knows or it can verify. If such a certificate is found, the VU shall use that certificate to verify the underlying card certificates it has read from the card. If successful, the process continues with the verification of the VU certificate chain. If not successful, the VU shall ignore the card.

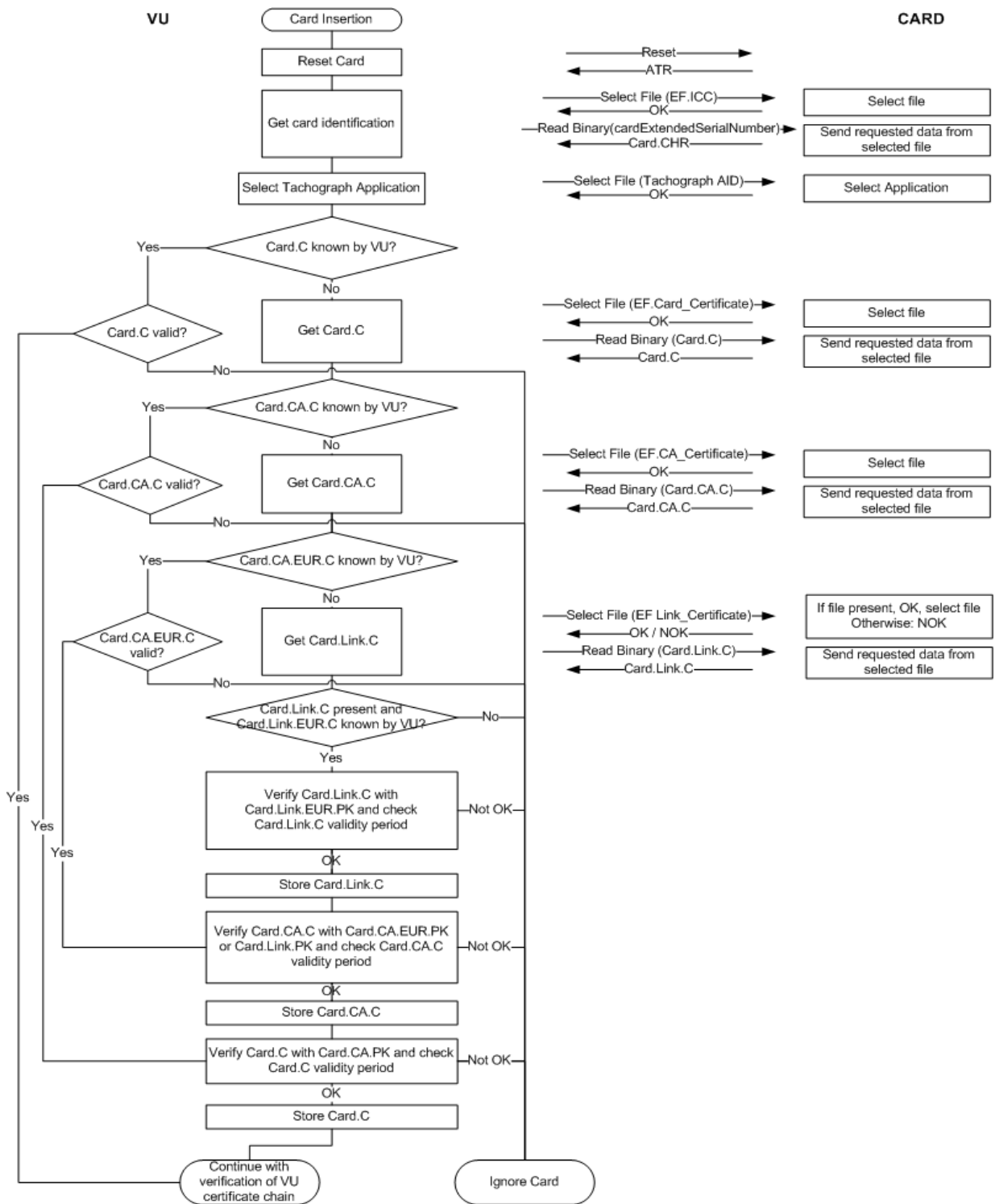
Note: There are three ways in which the VU may know the Card.CA.EUR certificate:

- the Card.CA.EUR certificate is the same certificate as the VU's own EUR certificate;
- the Card.CA.EUR certificate precedes the VU's own EUR certificate and the VU contained this certificate already at issuance (see CSM_81);
- the Card.CA.EUR certificate succeeds the VU's own EUR certificate and the VU received a link certificate in the past from another tachograph card, verified it and stored it for future reference.

CSM_159 As indicated in **Figure 4**, once the VU has verified the authenticity and validity of a previously unknown certificate, it may store this certificate for future reference, such that it does not need to verify that certificate's authenticity again if it is presented to the VU again. Instead of storing the entire certificate, a VU may choose to store only the contents of the Certificate Body, as specified in section **9.3.2**. **Whereas storing of all other types of certificate is optional, it is mandatory for a VU to store a new link certificate presented by a card.**

CSM_160 The VU shall verify the temporal validity of any certificate read from the card or stored in its memory, and shall reject expired certificates. For verifying the temporal validity of a certificate presented by the card a VU shall use its internal clock.

Figure 4
Protocol for Card Certificate Chain Verification by VU



10.2.2 VU Certificate Chain Verification by Card

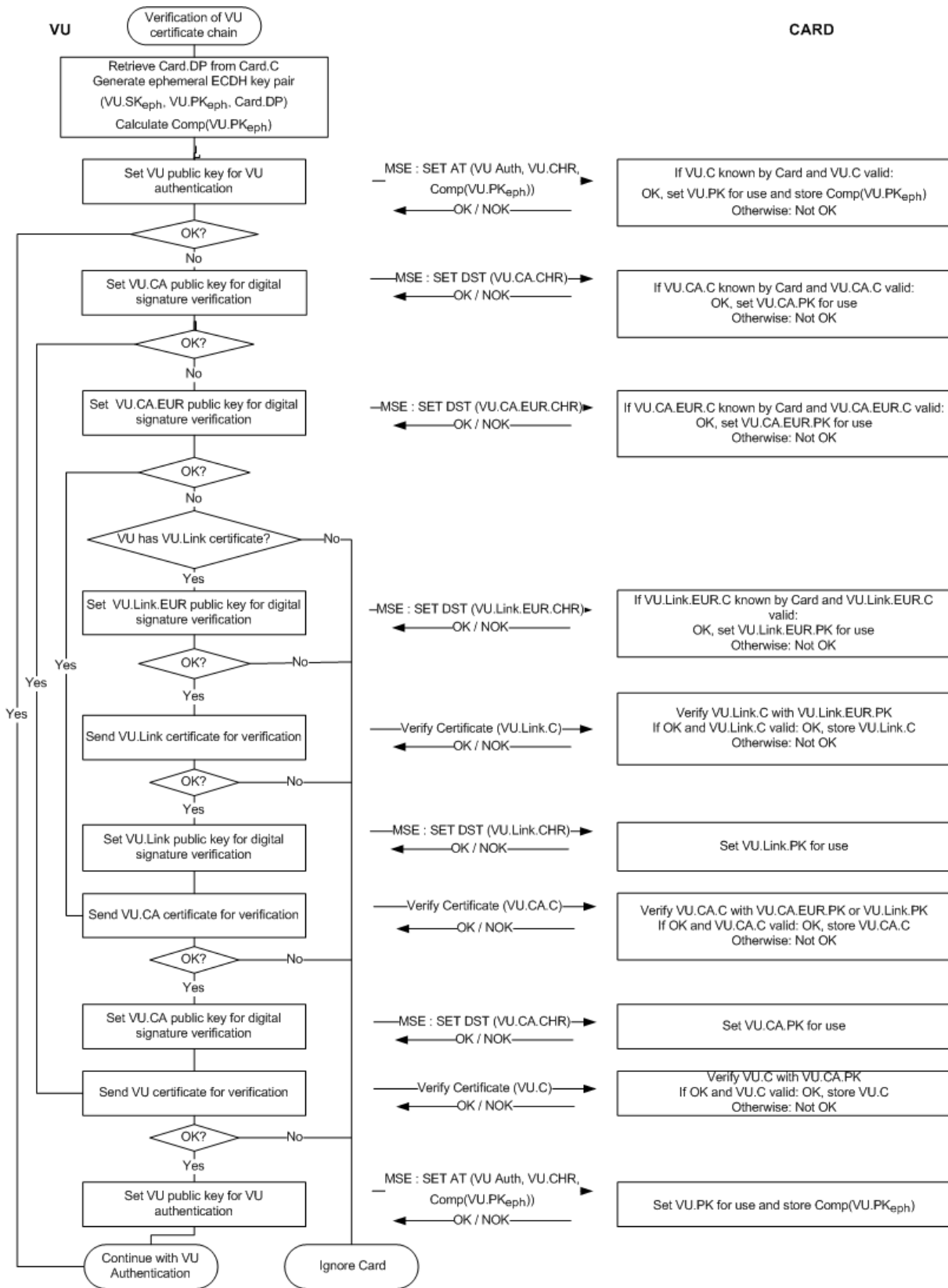
CSM_161 Tachograph cards shall use the protocol depicted in Figure 5 for verifying a VU's certificate chain. **For every certificate presented by the VU, the card shall verify that the Certificate Holder Authorisation (CHA) field is correct:**

- **The CHA of the VU.Link certificate shall indicate the ERCA.**
- **The CHA of the VU.CA certificate shall indicate an MSCA.**

Figure 5

Protocol for VU Certificate Chain Verification by Card

- **The CHA field of the VU certificate shall indicate a VU certificate for mutual authentication (see Sub-appendix 1, data type EquipmentType).**



Notes to Figure :

- The VU certificates and public keys mentioned in the figure are those for mutual authentication. Section 9.1.4 denotes these as VU_MA.
- The VU.CA certificates and public keys mentioned in the figure are those for signing VU and external GNSS facility certificates. Section 9.1.3 denotes these as MSCA_VU-EGF.
- The VU.CA.EUR certificate mentioned in the figure is the ~~European~~ root certificate that is indicated in the CAR of the VU.CA certificate.
- The VU.Link certificate mentioned in the figure is the VU's link certificate, if present. As specified in section 9.1.2, this is a link certificate for a new ~~European~~ root key pair created by the ERCA and signed by the previous ~~European root~~ private key.
- The VU.Link.EUR certificate is the ~~European~~ root certificate that is indicated in the CAR of the VU.Link certificate.

CSM_162 As depicted in **Figure 5**, verification of the certificate chain of the vehicle unit shall begin with the vehicle unit attempting to set its own public key for use in the tachograph card. If this succeeds, it means that the card successfully verified the VU's certificate chain in the past, and has stored the VU certificate for future reference. In this case, the VU certificate is set for use and the process continues with VU Authentication. If the card does not know the VU certificate, the VU shall successively present the VU.CA certificate to be used for verifying its VU certificate, the VU.CA.EUR certificate to be used for verifying the VU.CA certificate, and possibly the link certificate, in order to find a certificate known or verifiable by the card. If such a certificate is found, the card shall use that certificate to verify the underlying VU certificates presented to it. If successful, the VU shall finally set its public key for use in the tachograph card. If not successful, the VU shall ignore the card.

Note: There are three ways in which the card may know the VU.CA.EUR certificate:

- the VU.CA.EUR certificate is the same certificate as the card's own EUR certificate;
- the VU.CA.EUR certificate precedes the card's own EUR certificate and the card contained this certificate already at issuance (see CSM_91);
- the VU.CA.EUR certificate succeeds the card's own EUR certificate and the card received a link certificate in the past from another vehicle unit, verified it and stored it for future reference.

CSM_163 The VU shall use the MSE: Set AT command to set its public key for use in the tachograph card. As specified in ~~Appendix~~ **Sub-appendix 2**, this command contains an indication of the cryptographic mechanism that will be used with the key that is set. This mechanism shall be 'VU Authentication using the ECDSA algorithm, in combination with the hashing algorithm linked to the key size of the VU's VU_MA key pair, as specified in ~~TCS_192~~ CSM_50'.

CSM_164 The MSE: Set AT command also contains an indication of the ephemeral key pair which the VU will use during session key agreement (see section 10.4). Therefore, before sending the MSE: Set AT command, the VU shall generate an ephemeral ECC key pair. For generating the ephemeral key pair, the VU shall use the standardized domain parameters indicated in the card certificate. The ephemeral key pair is denoted as (VU.SK_{eph}, VU.PK_{eph}, Card.DP). The VU shall take the x-coordinate of the ECDH ephemeral public point as the key identification; this is called the compressed representation of the public key and denoted as Comp(VU.PK_{eph}).

CSM_165 If the MSE: Set AT command is successful, the card shall set the indicated VU.PK for subsequent use during Vehicle Authentication, and shall temporarily store

Comp(VU.PK_{eph}). In case two or more successful MSE: Set AT commands are sent before session key agreement is performed, the card shall store only the last Comp(VU.PK_{eph}) received. **The card shall reset Comp(VU.PK_{eph}) after a successful GENERAL AUTHENTICATE command.**

CSM_166 The card shall verify the temporal validity of any certificate presented by the VU or referenced by the VU while stored in the card's memory, and shall reject expired certificates.

CSM_167 For verifying the temporal validity of a certificate presented by the VU, each tachograph card shall internally store some data representing the current time. This data shall not be directly updatable by a VU. At issuance, the current time of a card shall be set equal to the Effective Date of the card's Card_MA certificate. A card shall update its current time if the Effective Date of an authentic 'valid source of time' certificate presented by a VU is more recent than the card's current time. In that case, the card shall set its current time to the Effective Date of that certificate. The card shall accept only the following certificates as a valid source of time:

- Second-generation ERCA link certificates
- Second-generation MSCA certificates
- Second-generation VU certificates issued by the same country as the card's own card certificate(s).

Note: the last requirement implies that a card shall be able to recognize the CAR of the VU certificate, i.e. the MSCA_VU-EGF certificate. This will not be the same as the CAR of its own certificate, which is the MSCA_Card certificate.

CSM_168 As indicated in **Figure 5**, once the card has verified the authenticity and validity of a previously unknown certificate, it may store this certificate for future reference, such that it does not need to verify that certificate's authenticity again if it is presented to the card again. Instead of storing the entire certificate, a card may choose to store only the contents of the Certificate Body, as specified in section 9.3.2.

10.3. VU Authentication

CSM_169 Vehicle units and cards shall use the VU Authentication protocol depicted in **Figure 6** to authenticate the VU towards the card. VU Authentication enables the tachograph card to explicitly verify that the VU is authentic. To do so, the VU shall use its private key to sign a challenge generated by the card.

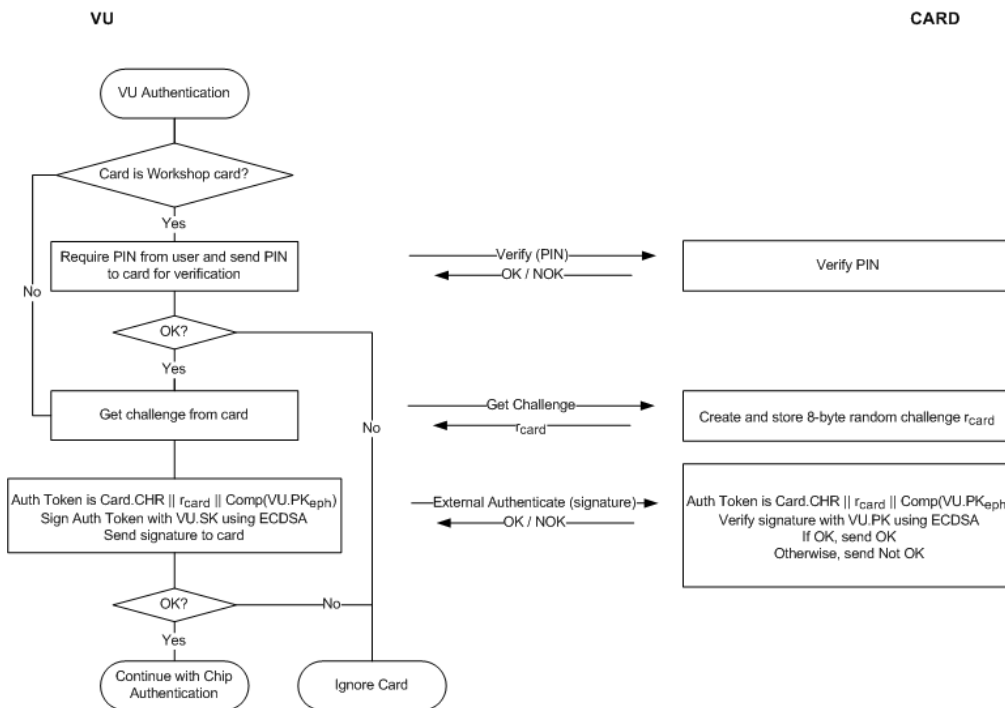
CSM_170 Next to the card challenge, the VU shall include in the signature the ~~card~~ **certificate** holder reference taken from the card certificate.

Note: This ensures that the card to which the VU authenticates itself is the same card whose certificate chain the VU has verified previously.

CSM_171 The VU shall also include in the signature the identifier of the ephemeral public key Comp(VU.PK_{eph}) which the VU will use to set up Secure Messaging during the Chip Authentication process specified in section **10.4**.

Note: This ensures that the VU with which a card communicates during a Secure Messaging session is the same VU that was authenticated by the card.

Figure 6
VU Authentication protocol



CSM_172 If multiple GET CHALLENGE commands are sent by the VU during VU Authentication, the card shall return a new 8-byte random challenge each time, but shall store only the last challenge.

CSM_173 The signing algorithm used by the VU for VU Authentication shall be ECDSA as specified in [DSS], using the hashing algorithm linked to the key size of the VU’s VU_MA key pair, as specified in TCS_192 CSM_50. The signature format shall be plain, as specified in [TR-03111]. The VU shall send the resulting signature to the card.

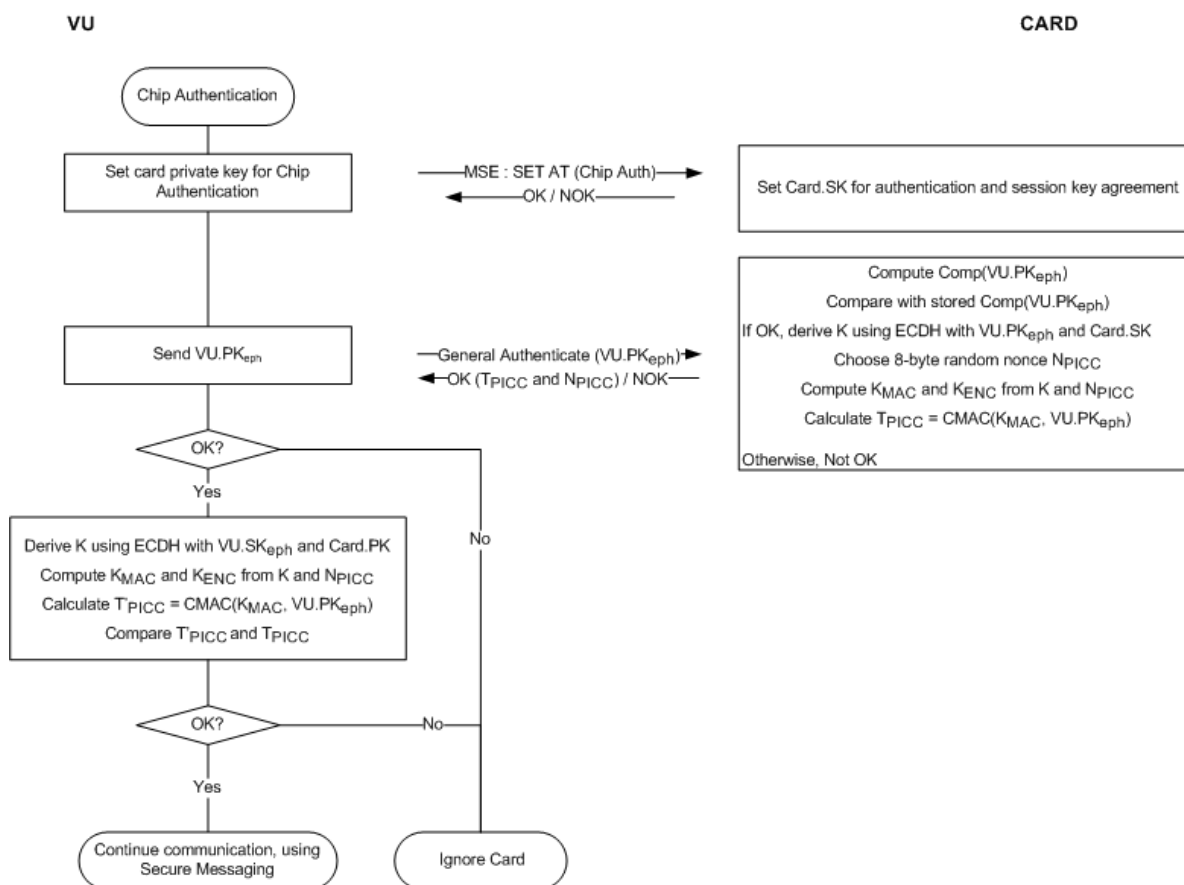
CSM_174 Upon receiving the VU’s signature in an EXTERNAL AUTHENTICATE command, the card shall

- Calculate the authentication token by concatenating Card.CHR, the card challenge card and the identifier of the VU ephemeral public key Comp(VU.PKeph),
- Calculate Verify the hash over the authentication token VU’s signature using the ECDSA algorithm the hashing algorithm linked to the key size of the VU’s VU_MA key pair as specified in TCS_192
- Verify the VU’s signature using the ECDSA algorithm CSM_50, in combination with VU.SK and the calculated hash authentication token.

10.4. Chip Authentication and Session Key Agreement

CSM_175 Vehicle units and cards shall use the Chip Authentication protocol depicted in Figure 7 to authenticate the card towards the VU. Chip Authentication enables the vehicle unit to explicitly verify that the card is authentic.

Figure 7
Chip Authentication and session key agreement



CSM_176 The VU and the card shall take the following steps:

1. The vehicle unit initiates the Chip Authentication process by sending the MSE: Set AT command indicating 'Chip Authentication using the ECDH algorithm resulting in an AES session key length linked to the key size of the card's Card_MA key pair, as specified in ~~TCS_192~~ **CSM_50**'. The VU shall determine the key size of the card's key pair from the card certificate.
2. The VU sends the public point VU.PK_{eph} of its ephemeral key pair to the card. **The public point shall be converted to an octet string as specified in [TR-03111]. The uncompressed encoding format shall be used.** As explained in ~~TCS_306~~ **CSM_164**, the VU generated this ephemeral key pair prior to the verification of the VU certificate chain. The VU sent the identifier of the ephemeral public key Comp(VU.PK_{eph}) to the card, and the card stored it.
3. The card computes Comp(VU.PK_{eph}) from VU.PK_{eph} and compares this to the stored value of Comp(VU.PK_{eph}).
4. Using the ECDH algorithm in combination with the card's static private key and the VU's ephemeral public key, the card computes a secret K.
5. The card chooses a random 8-byte nonce NP_{ICC} and uses it to derive two AES session keys K_{MAC} and K_{ENC} from K. See ~~TCS_321~~ **CSM_179**.
6. Using K_{MAC}, the card computes an authentication token over the VU ephemeral public key identifier: TP_{ICC} = CMAC(K_{MAC}, VU.PK_{eph}). **The public point shall**

be in the format used by the VU (see bullet 2 above). The card sends NPICC and TPICC to the vehicle unit.

7. Using the ECDH algorithm in combination with the card's static public key and the VU's ephemeral private key, the VU computes the same secret K as the card did in step 30.
8. The VU derives session keys K_{MAC} and K_{ENC} from K and NPICC; see ~~TCS_324~~ **CSM_179**.
9. The VU verifies the authentication token TPICC.

CSM_177 In step 2 above, the card shall compute $\text{Comp}(\text{VU.PKeph})$ as the x-coordinate of the public point in VU.PKeph.

CSM_178 In steps 3 and 6 above, the card and the vehicle unit shall use the ECKA-EG algorithm as defined in [TR-03111].

CSM_179 In steps 4 and 7 above, the card and the vehicle unit shall use the key derivation function for AES session keys defined in [TR-03111], with the following precisions and changes:

- The value of the counter shall be '00 00 00 01' for K_{ENC} and '00 00 00 02' for K_{MAC} .
- The optional nonce r shall be used and shall be equal to NPICC.
- For deriving 128-bits AES keys, the hashing algorithm to be used shall be SHA-256.
- For deriving 192-bits AES keys, the hashing algorithm to be used shall be SHA-384.
- For deriving 256-bits AES keys, the hashing algorithm to be used shall be SHA-512.

The length of the session keys (i.e. the length at which the hash is truncated) shall be linked to the size of the Card_MA key pair, as specified in ~~TCS_192~~ **CSM_50**.

CSM_180 In steps 6 and 9 above, the card and the vehicle unit shall use the AES algorithm in CMAC mode, as specified in [SP 800-38B]. The length of TPICC shall be linked to the length of the AES session keys, as specified in ~~TCS_192~~ **CSM_50**.

10.5. Secure Messaging

10.5.1 General

CSM_181 All commands and responses exchanged between a vehicle unit and a tachograph card after successful Chip Authentication took place and until the end of the session shall be protected by Secure Messaging.

CSM_182 Except when reading from a file with access condition SM-R-ENC-MAC-G2 (see ~~Appendix~~ **Sub-appendix** 2, section 4), Secure Messaging shall be used in authentication-only mode. In this mode, a cryptographic checksum (a.k.a. MAC) is added to all commands and responses to ensure message authenticity and integrity.

CSM_183 When reading data from a file with access condition SM-R-ENC-MAC-G2, Secure Messaging shall be used in encrypt-then-authenticate mode, i.e. the response data is encrypted first to ensure message confidentiality, and afterwards a MAC over the formatted encrypted data is calculated to ensure authenticity and integrity.

CSM_184 Secure Messaging shall use AES as defined in [AES] with the session keys K_{MAC} and K_{ENC} that were agreed during Chip Authentication.

CSM_185 An unsigned integer shall be used as the Send Sequence Counter (SSC) to prevent replay attacks. The size of the SSC shall be equal to the AES block size, i.e. 128 bits. The SSC shall be in MSB-first format. The Send Sequence Counter shall be initialized to zero (i.e. '00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00') when Secure Messaging is started. The SSC shall be increased every time before a command or response APDU is generated, i.e. since the starting value of the SSC in a SM session is 0, in the first command the value of the SSC will be 1. The value of SSC for the first response will be 2.

CSM_186 For message encryption, K_{ENC} shall be used with AES in the Cipher Block Chaining (CBC) mode of operation, as defined in [ISO 10116], with an interleave parameter $m = 1$ and an initialization vector $SV = E(K_{ENC}, SSC)$, i.e. the current value of the Send Sequence Counter encrypted with K_{ENC} .

CSM_187 For message authentication, K_{MAC} shall be used with AES in CMAC mode as specified in [SP 800-38B]. The length of the MAC shall be linked to the length of the AES session keys, as specified in ~~TCS_192~~ CSM_50. The Send Sequence Counter shall be included in the MAC by prepending it before the datagram to be authenticated.

10.5.2 Secure Message Structure

CSM_188 Secure Messaging shall make use only of the Secure Messaging data objects (see [ISO 7816-4]) listed in **Table 547**. In any message, these data objects shall be used in the order specified in this table.

Table5

Secure Messaging Data Objects

Data Object Name	Tag	Presence (M)andatory, (C)onditional or (F)orbidden in	
		Commands	Responses
Plain value not encoded in BER-TLV	'81'	C	C
Plain value encoded in BER-TLV, but not including SM DOs	'B3'	C	C
Padding-content indicator followed by cryptogram, plain value not encoded in BER-TLV	'87'	C	C
Protected Le	'97'	C	F
Processing Status	'99'	F	M
Cryptographic Checksum	'8E'	M	M

Note: As specified in **Appendix Sub-appendix 2**, tachograph cards may support the READ BINARY and UPDATE BINARY command with an odd INS byte ('B1' resp. 'D7'). These command variants are required to read and update files with more than 32768 bytes or more. In case such a variant is used, a data object with tag 'B3' shall be used instead of an object with tag '81'. See **Appendix Sub-appendix 2** for more information.

CSM_189 All SM data objects shall be encoded in DER TLV as specified in [ISO 8825-1]. This encoding results in a Tag-Length-Value (TLV) structure as follows:

Tag: ~~T~~the tag is encoded in one or two octets and indicates the content.

Length: ~~T~~the length is encoded as an unsigned integer in one, two, or three octets, resulting in a maximum length of 65535 octets. The minimum number of octets shall be used.

Value: ~~T~~the value is encoded in zero or more octets

CSM_190 APDUs protected by Secure Messaging shall be created as follows:

- The command header shall be included in the MAC calculation, therefore value '0C' shall be used for the class byte CLA.
- As specified in ~~Appendix~~ **Sub-appendix 2**, all INS bytes shall be even, with the possible exception of odd INS bytes for the READ BINARY and UPDATE BINARY commands.
- The actual value of Lc will be modified to Lc' after application of secure messaging.
- The Data field shall consist of SM data objects.
- In the protected command APDU the new Le byte shall be set to '00'. If required, a data object '97' shall be included in the Data field in order to convey the original value of Le.

CSM_191 Any data object to be encrypted shall be padded according to [ISO 7816-4] using padding-content indicator '01'. For the calculation of the MAC, ~~each data object~~ **objects** in the APDU shall ~~also be separately~~ padded according to [ISO 7816-4].

Note: Padding for Secure Messaging is always performed by the secure messaging layer, not by the CMAC or CBC algorithms.

Summary and Examples

A command APDU with applied Secure Messaging will have the following structure, depending on the case of the respective unsecured command (DO is data object):

Case 1:	CLA INS P1 P2 Lc' DO '8E' Le
Case 2:	CLA INS P1 P2 Lc' DO '97' DO'8E' Le
Case 3 (even INS byte):	CLA INS P1 P2 Lc' DO '81' DO'8E' Le
Case 3 (odd INS byte):	CLA INS P1 P2 Lc' DO 'B3' DO'8E' Le
Case 4 (even INS byte):	CLA INS P1 P2 Lc' DO '81' DO'97' DO'8E' Le
Case 4 (odd INS byte):	CLA INS P1 P2 Lc' DO 'B3' DO'97' DO'8E' Le

where Le = '00' or '00 00' depending on whether short length fields or extended length fields are used; see [ISO 7816-4].

A response APDU with applied Secure Messaging will have the following structure, depending on the case of the respective unsecured response:

Case 1 or 3:	DO '99' DO '8E' SW1SW2
Case 2 or 4 (even INS byte) with encryption:	DO '81' DO '99' DO '8E' SW1SW2
Case 2 or 4 (even INS byte) without encryption:	DO '87' DO '99' DO '8E' SW1SW2
Case 2 or 4 (odd INS byte) without encryption:	DO 'B3' DO '99' DO '8E' SW1SW2

Note: Case 2 or 4 (odd INS byte) with encryption is never used in the communication between a VU and a card.

Below are three example APDU transformations for commands with even INS code. **Figure 8** shows an authenticated Case 4 command APDU, **Figure 9** shows an authenticated Case 1/Case 3 response APDU, and **Figure 10** shows an encrypted and authenticated Case 2/Case 4 response APDU.

Figure 8
Transformation of an authenticated Case 4 Command APDU

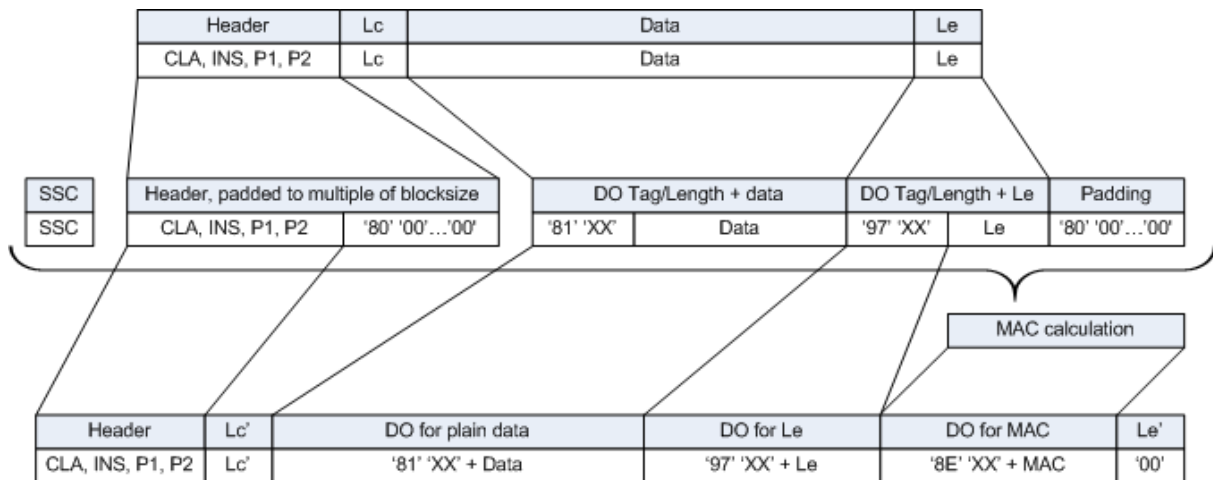


Figure 9
Transformation of an authenticated Case 1 / Case 3 Response APDU

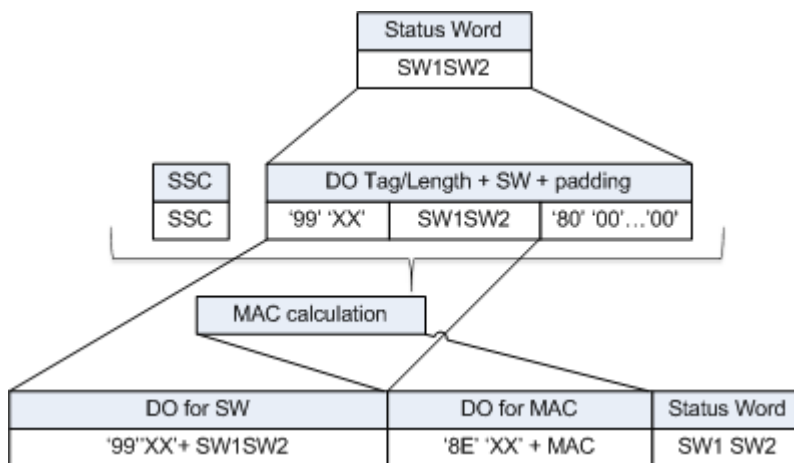
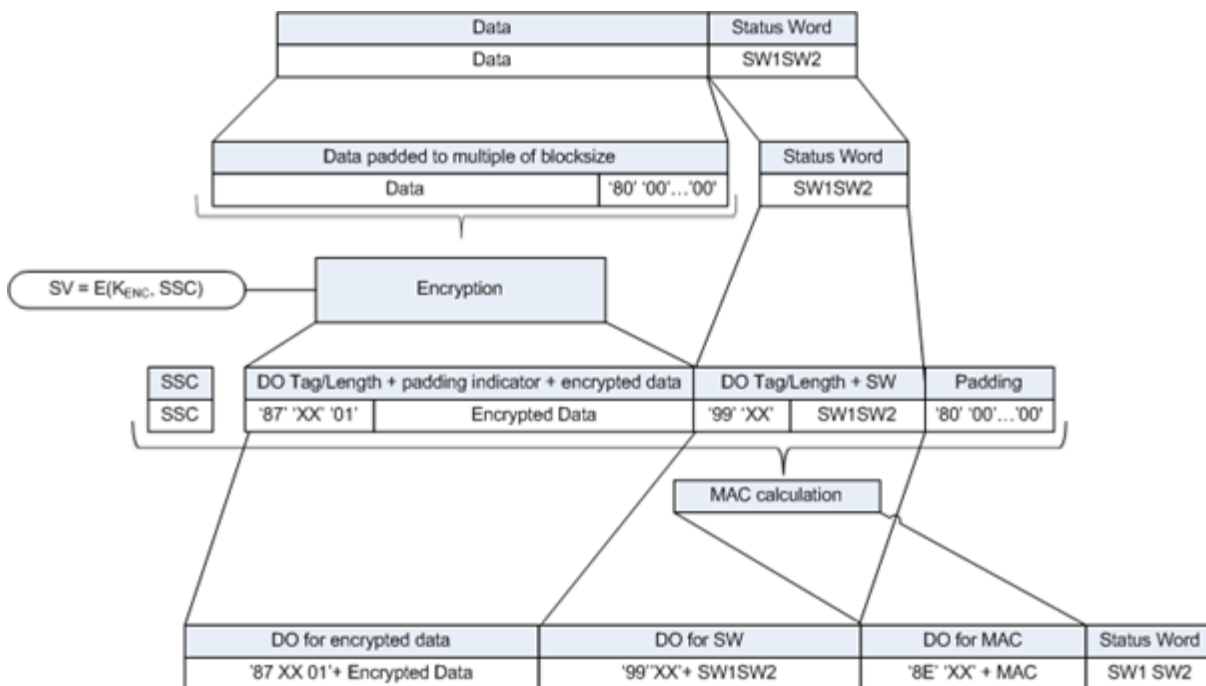


Figure 10
Transformation of an encrypted and authenticated Case 2/Case 4 Response APDU



10.5.3 Secure Messaging Session Abortion

CSM_192 A vehicle unit shall abort an ongoing Secure Messaging session if and only if one of the following conditions occur:

- it receives a plain response APDU,
- it detects a Secure Messaging error in a response APDU:
 - An expected Secure Messaging data object is missing, the order of data objects is incorrect, or an unknown data object is included.
 - A Secure Messaging data object is incorrect, e.g. the MAC value is incorrect, the TLV structure is incorrect or the padding indicator in tag '87' is not equal to '01'.
- the card sends a status byte indicating it detected an SM error (see ~~TCS_336~~ **CSM_194**),
- the limit for the number of commands and associated responses within the current session is reached. For a given VU, this limit shall be defined by its manufacturer, taking into account the security requirements of the hardware used, with a maximum value of 240 SM commands and associated responses per session.

CSM_193 A tachograph card shall abort an ongoing Secure Messaging session if and only if one of the following conditions occur:

- it receives a plain command APDU,
- it detects a Secure Messaging error in a command APDU:
 - An expected Secure Messaging data object is missing, the order of data objects is incorrect, or an unknown data object is included.

- A Secure Messaging data object is incorrect, e.g. the MAC value is incorrect or the TLV structure is incorrect.
- it is depowered or reset,
- ~~— the VU selects an application on the card~~
- the VU starts the VU Authentication process,
- the limit for the number of commands and associated responses within the current session is reached. For a given card, this limit shall be defined by its manufacturer, taking into account the security requirements of the hardware used, with a maximum value of 240 SM commands and associated responses per session.

CSM_194 Regarding SM error handling by a tachograph card:

- If in a command APDU some expected Secure Messaging data objects are missing, the order of data objects is incorrect or unknown data objects are included, a tachograph card shall respond with status bytes '69 87'.
- If a Secure Messaging data object in a command APDU is incorrect, a tachograph card shall respond with status bytes '69 88'.

In such a case, the status bytes shall be returned without using SM.

CSM_195 If a Secure Messaging session between a VU and a tachograph card is aborted, the VU and the tachograph card shall

- securely destroy the stored session keys
- immediately establish a new Secure Messaging session, as described in sections **10.2 – 10.5**.

CSM_196 If for any reason the VU decides to restart mutual authentication towards an inserted card, the process shall restart with verification of the card certificate chain, as described in section **10.2**, and shall continue as described in sections **10.2 – 10.5**.

11. VU – External GNSS Facility Coupling, Mutual Authentication and Secure Messaging

11.1. General

CSM_197 The GNSS facility used by a VU to determine its position may be internal, (i.e. built into the VU casing and not detachable), or it may be an external module. In the first case, there is no need to standardize the internal communication between the GNSS facility and the VU, and the requirements in this chapter do not apply. In the latter case, communication between the VU and the external GNSS facility shall be standardized and protected as described in this chapter.

CSM_198 Secure communication between a vehicle unit and an external GNSS facility shall take place in the same way as secure communication between a vehicle unit and a tachograph card, with the external GNSS facility (EGF) taking the role of the card. All requirements mentioned in chapter **10** for tachograph cards shall be satisfied by an EGF, taking into account the deviations, clarifications and additions mentioned in this chapter. In particular, mutual certificate chain verification, VU Authentication and Chip Authentication shall be performed as described in sections **11.3 and 11.4**.

CSM_199 Communication between a vehicle unit and an EGF differs from communication between a vehicle unit and a card in the fact that a vehicle unit and an EGF

must be coupled once in a workshop before the VU and the EGF can exchange GNSS-based data during normal operation. The coupling process is described in section **11.2**.

CSM_200 For communication between a vehicle unit and an EGF, APDU commands and responses based on [ISO 7816-4] and [ISO 7816-8] shall be used. The exact structure of these APDUs is defined in ~~Appendix~~ **Sub-appendix 2** of this ~~Annex~~ **Appendix**.

11.2. VU and External GNSS Facility Coupling

CSM_201 A vehicle unit and an EGF in a vehicle shall be coupled by a workshop. Only a coupled vehicle unit and EGF shall be able to communicate during normal operation.

CSM_202 Coupling of a vehicle unit and an EGF shall only be possible if the vehicle unit is in calibration mode. The coupling shall be initiated by the vehicle unit.

CSM_203 A workshop may re-couple a vehicle unit to another EGF or to the same EGF at any time. During re-coupling, the VU shall securely destroy the existing EGF_MA certificate in its memory and shall store the EGF_MA certificate of the EGF to which it is being coupled.

CSM_204 A workshop may re-couple an external GNSS facility to another VU or to the same VU at any time. During re-coupling, the EGF shall securely destroy the existing VU_MA certificate in its memory and shall store the VU_MA certificate of the VU to which it is being coupled.

11.3. Mutual Certificate Chain Verification

11.3.1 General

CSM_205 Mutual certificate chain verification between a VU and an EGF shall take place only during the coupling of the VU and the EGF by a workshop. During normal operation of a coupled VU and EGF, no certificates shall be verified. Instead, the VU and EGF shall trust the certificates they stored during the coupling, after checking the temporal validity of these certificates. The VU and the EGF shall not trust any other certificates for protecting the VU – EGF communication during normal operation.

11.3.2 During VU – EGF Coupling

CSM_206 During the coupling to an EGF, a vehicle unit shall use the protocol depicted in **Figure 4** (section 10.2.1) for verifying the external GNSS facility's certificate chain.

Notes to **Figure 4** within this context:

- Communication control is out of the scope of this ~~Appendix~~ **sub-appendix**. However, an EGF is not a smart card and hence the VU will probably not send a Reset to initiate the communication and will not receive an ATR.
- The Card certificates and public keys mentioned in the figure shall be interpreted as the EGF's certificates and public keys for mutual authentication. Section **9.1.6** denotes these as EGF_MA.
- The Card.CA certificates and public keys mentioned in the figure shall be interpreted as the MSCA's certificates and public keys for signing EGF certificates. Section **9.1.3** denotes these as MSCA_VU-EGF.
- The Card.CA.EUR certificate mentioned in the figure shall be interpreted as the ~~European~~ root certificate that is indicated in the CAR of the MSCA_VU-EGF certificate.

- The Card.Link certificate mentioned in the figure shall be interpreted as the EGF's link certificate, if present. As specified in section 9.1.2, this is a link certificate for a new ~~European~~ root key pair created by the ERCA and signed by the previous ~~European root~~ private key.
- The Card.Link.EUR certificate is the ~~European~~ root certificate that is indicated in the CAR of the Card.Link certificate.
- Instead of the cardExtendedSerialNumber, the VU shall read the sensorGNSSserialNumber from EF ICC.
- Instead of selecting the Tachograph AID, the VU shall select the EGF AID.
- 'Ignore Card' shall be interpreted as 'Ignore EGF'.

CSM_207 Once it has verified the EGF_MA certificate, the vehicle unit shall store this certificate for use during normal operation; see section 11.3.3.

CSM_208 During the coupling to a VU, an external GNSS ~~unit~~ facility shall use the protocol depicted in Figure 5 (section 10.2.2) for verifying the VU's certificate chain.

Notes to **Figure 5** within this context:

- The VU shall generate a fresh ephemeral key pair using the domain parameters in the EGF certificate.
- The VU certificates and public keys mentioned in the figure are those for mutual authentication. Section 9.1.4 denotes these as VU_MA.
- The VU.CA certificates and public keys mentioned in the figure are those for signing VU and external GNSS facility certificates. Section 9.1.3 denotes these as MSCA_VU-EGF.
- The VU.CA.EUR certificate mentioned in the figure is the ~~European~~ root certificate that is indicated in the CAR of the VU.CA certificate.
- The VU.Link certificate mentioned in the figure is the VU's link certificate, if present. As specified in section 9.1.2, this is a link certificate for a new ~~European~~ root key pair created by the ERCA and signed by the previous ~~European root~~ private key.
- The VU.Link.EUR certificate is the ~~European~~ root certificate that is indicated in the CAR of the VU.Link certificate.

CSM_209 In deviation from requirement ~~TCS_309~~ CSM_167, an EGF shall use the GNSS time to verify the temporal validity of any certificate presented.

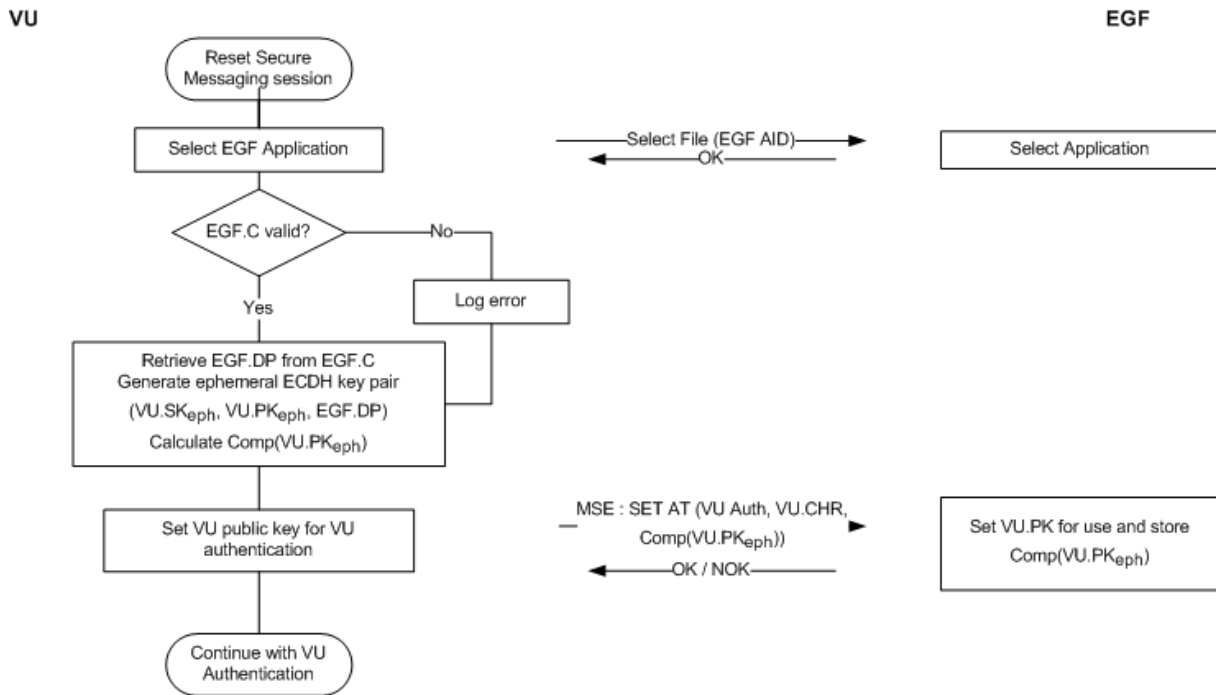
CSM_210 Once it has verified the VU_MA certificate, the external GNSS ~~unit~~ facility shall store this certificate for use during normal operation; see section 11.3.3.

11.3.3 During Normal Operation

CSM_211 During normal operation, a vehicle unit and an EGF shall use the protocol depicted in Figure 11 for verifying the temporal validity of the stored EGF ~~and MA and VU_MA certificates~~ and for setting the VU_MA public key for subsequent VU Authentication. No further mutual verification of the certificate chains shall take place during normal operation.

Note that **Figure 11** in essence consists of the first steps shown in **Figure 4** and **Figure 5**. Again, note that since an EGF is not a smart card, the VU will probably not send a Reset to initiate the communication and will not receive an ATR. In any case this is out of the scope of this ~~Appendix~~ sub-appendix.

Figure 11
Mutual verification of certificate temporal validity during normal VU - EGF operation



CSM_212 As shown in **Figure 11**, the vehicle unit shall log an error if the EGF_MA certificate is no longer valid. However, mutual authentication, key agreement and subsequent communication via secure messaging shall proceed normally.

11.4. VU Authentication, Chip Authentication and Session Key Agreement

CSM_213 VU Authentication, Chip Authentication and session key agreement between a VU and an EGF shall take place during coupling and whenever a Secure Messaging session is re-established during normal operation. The VU and the EGF shall carry out the processes described in sections **10.3 and 10.4**. All requirements in these sections shall apply.

11.5. Secure Messaging

CSM_214 All commands and responses exchanged between a vehicle unit and an external GNSS facility after successful Chip Authentication took place and until the end of the session shall be protected by Secure Messaging in authentication-only mode. All requirements in section **10.5** shall apply.

CSM_215 If a Secure Messaging session between a VU and an EGF is aborted, the VU shall immediately establish a new Secure Messaging session, as described in section **11.3.3 and 11.4**.

12. VU – Motion Sensor Pairing and Communication

12.1. General

CSM_216 A vehicle unit and a motion sensor shall communicate using the interface protocol specified in [ISO 16844-3] during pairing and in normal operation, with the changes described in this chapter and in section 9.2.1.

Note: readers of this chapter are supposed to be familiar with the contents of [ISO 16844-3].

12.2. VU – Motion Sensor Pairing Using Different Key Generations

As explained in section 9.2.1, the motion sensor master key and all associated keys are regularly replaced. This leads to the presence of up to three motion sensor-related AES keys K_{M-WC} (of consecutive key generations) in workshop cards. Similarly, in motion sensors up to three different AES-based encryptions of data (based on consecutive generations of the motion sensor master key K_M) may be present. A vehicle unit contains only one motion sensor-related key K_{M-VU} .

CSM_217 A second-generation VU and a second-generation motion sensor shall be paired as follows (compare Table 6 in [ISO 16844-3]):

1. A second-generation workshop card is inserted into the VU and the VU is connected to the motion sensor.
2. The VU reads all available K_{M-WC} keys from the workshop card, inspects their key version numbers and chooses the one matching the version number of the VU's K_{M-VU} key. If the matching K_{M-WC} key is not present on the workshop card, the VU aborts the pairing process and shows an appropriate error message to the workshop card holder.
3. The VU calculates the motion sensor master key K_M from K_{M-VU} and K_{M-WC} , and the identification key K_{ID} from K_M , as specified in section 9.2.1.
4. The VU sends the instruction to initiate the pairing process towards the motion sensor, as described in [ISO 16844-3], and encrypts the serial number it receives from the motion sensor with the identification key K_{ID} . The VU sends the encrypted serial number back to the motion sensor.
5. The motion sensor matches the encrypted serial number consecutively with each of the encryptions of the serial number it holds internally. If it finds a match, the VU is authenticated. The motion sensor notes the generation of K_{ID} used by the VU and returns the matching encrypted version of its pairing key; i.e. the encryption that was created using the same generation of K_M .
6. The VU decrypts the pairing key using K_M , generates a session key K_S , encrypts it with the pairing key and sends the result to the motion sensor. The motion sensor decrypts K_S .
7. The VU assembles the pairing information as defined in [ISO 16844-3], encrypts the information with the pairing key, and sends the result to the motion sensor. The motion sensor decrypts the pairing information.
8. The motion sensor encrypts the received pairing information with the received K_S and returns this to the VU. The VU verifies that the pairing information is the same information which the VU sent to the motion sensor in the previous step. If it is, this proves that the motion sensor used the same K_S as the VU and hence in step 5 sent

its pairing key encrypted with the correct generation of K_M . Hence, the motion sensor is authenticated.

Note that steps 2 and 5 are different from the standard process in [ISO 16844-3]; the other steps are standard.

Example: Suppose a pairing takes place in the first year of the validity of the ERCA (3) certificate; see Figure 2 in section 9.2.2.2-9.2.1.2. Moreover

- Suppose the motion sensor was issued in the last year of the validity of the ERCA (1) certificate. It will therefore contain the following keys and data:
- $N_s[1]$: its serial number encrypted with generation 1 of K_{ID} ,
- $N_s[2]$: its serial number encrypted with generation 2 of K_{ID} ,
- $N_s[3]$: its serial number encrypted with generation 3 of K_{ID} ,
- $K_P[1]$: its generation-1 pairing key¹³, encrypted with generation 1 of K_M ,
- $K_P[2]$: its generation-2 pairing key, encrypted with generation 2 of K_M ,
- $K_P[3]$: its generation-3 pairing key, encrypted with generation 3 of K_M ,
- Suppose that the workshop card was issued in the first year of the validity of the ERCA (3) certificate. It will therefore contain the generation 2 and generation 3 of the K_{M-WC} key.
- Suppose the VU is a generation-2 VU, containing the generation 2 of K_{M-VU} .
- In this case, the following will happen in steps 2 - 5:
- Step 2: The VU reads generation 2 and generation 3 of K_{M-WC} from the workshop card and inspects their version numbers.
- Step 3: The VU combines the generation-2 K_{M-WC} with its K_{M-VU} to compute K_M and K_{ID} .
- Step 4: The VU encrypts the serial number it receives from the motion sensor with K_{ID} .
- Step 5: The motion sensor compares the received data with $N_s[1]$ and doesn't find a match. Next, it compares the data with $N_s[2]$ and finds a match. It concludes that the VU is a generation-2 VU, and therefore sends back $K_P[2]$.

12.3. VU – Motion Sensor Pairing and Communication using AES

CSM_218 As specified in Table 345 in section 9.2.1, all keys involved in the pairing of a (second-generation) vehicle unit and a motion sensor and in subsequent communication shall be AES keys, rather than double-length TDES keys as specified in [ISO 16844-3]. These AES keys may have a length of 128, 192 or 256 bits. Since the AES block size is 16 bytes, the length of an encrypted message must be a multiple of 16 bytes, compared to 8 bytes for TDES. Moreover, some of these messages will be used to transport AES keys, the length of which may be 128, 192 or 256 bits. Therefore, the number of data bytes per instruction in Table 5 of [ISO 16844-3] shall be changed as shown in Table 648:

¹³ Note that the generation-1, generation-2 and generation-3 pairing keys may actually be the same key, or may be three different keys having different lengths, as explained in TCS_259 CSM_117.

Table 6
Number of plaintext and encrypted data bytes per instruction defined in [ISO 16844-3]

Instruction	Request / reply	Description of data	# of plaintext data bytes according to [ISO 16844-3]	# of plaintext data bytes using AES keys	# of encrypted data bytes when using AES keys of bitlength		
					128	192	256
10	request	Authentication data + file number	8	8	16	16	16
11	reply	Authentication data + file contents	16 or 32, depend on file	16 or 32, depend on file	16 32 / 48	16 32 / 48	16 32 / 48
41	request	MoS serial number	8	8	16	16	16
41	reply	Pairing key	16	16 / 24 / 32	16	32	32
42	request	Session key	16	16 / 24 / 32	16	32	32
43	request	Pairing information	24	24	32	32	32
50	reply	Pairing information	24	24	32	32	32
70	request	Authentication data	8	8	16	16	16
80	reply	MoS counter value + auth. data	8	8	16	16	16

CSM_219 The pairing information that is sent in instructions 43 (VU request) and 50 (MoS reply) shall be assembled as specified in section 7.6.10 of [ISO 16844-3], except that the AES algorithm shall be used instead of the TDES algorithm in the pairing data encryption scheme, thus resulting in two AES encryptions, and adopting the padding specified in ~~TCS_362~~ CSM_220 to fit with the AES block size. The key K_p^* used for this encryption shall be generated as follows:

- In case the pairing key K_p is 16 bytes long: $K_p^* = K_p \text{ XOR } (N_s || N_s)$
- In case the pairing key K_p is 24 bytes long: $K_p^* = K_p \text{ XOR } (N_s || N_s || N_s)$
- In case the pairing key K_p is 32 bytes long: $K_p^* = K_p \text{ XOR } (N_s || N_s || N_s || N_s)$

where N_s is the 8-byte serial number of the motion sensor.

CSM_220 In case the plaintext data length (using AES keys) is not a multiple of 16 bytes, padding method 2 defined in [ISO 9797-1] shall be used.

Note: in [ISO 16844-3], the number of plaintext data bytes is always a multiple of 8, such that padding is not necessary when using TDES. The definition of data and messages in [ISO 16844-3] is not changed by this part of this ~~Appendix~~ **sub-appendix**, thus necessitating the application of padding.

CSM_221 For instruction 11 and in case more than one block of data must be encrypted, the Cipher Block Chaining mode of operation shall be used as defined in [ISO 10116], with an interleave parameter $m = 1$. The IV to be used shall be

- For instruction 11: the 8-byte authentication block specified in section 7.6.3.3 of [ISO 16844-3], padded using padding method 2 defined in [ISO 9797-1]; see also section 7.6.5 and 7.6.6 of [ISO 16844-3].
- For all other instructions in which more than 16 bytes are transferred, as specified in **Table 648**: '00' {16}, i.e. sixteen bytes with binary value 0.

Note: As shown in section 7.6.5 and 7.6.6 of [ISO 16844-3], when the MoS encrypts data files for inclusion in instruction 11, the authentication block is both

- Used as the initialization vector for the CBC-mode encryption of the data files
- Encrypted and included as the first block in the data that is sent to the VU.

12.4. VU – Motion Sensor Pairing For Different Equipment Generations

CSM_222 As explained in section 9.2.1, a second-generation motion sensor may contain the ~~TDES-based~~ encryption of the pairing data (as defined in Part A of this ~~Appendix-sub-appendix~~), which allows the motion sensor to be paired to a first-generation VU. If this is the case, a ~~first-generation VU and a second-generation motion sensor~~ shall be paired as described in Part A of this ~~Appendix-sub-appendix~~ and in [ISO 16844-3]. For the pairing process either a first-generation or a second-generation workshop card may be used.

Notes:

- It is not possible to pair a second-generation VU to a first-generation motion sensor.
- It is not possible to use a first-generation workshop card for coupling a second-generation VU to a motion sensor.

13. Security for Remote Communication over DSRC

13.1. General

As specified in ~~Appendix-Sub-appendix~~ 14, a VU regularly generates Remote Tachograph Monitoring (RTM) data and sends this data to the (internal or external) Remote Communication Facility (RCF). The remote communication facility is responsible for sending this data over the DSRC interface described in ~~Appendix-Sub-appendix~~ 14 to the remote interrogator. ~~Appendix-Sub-appendix~~ 1 specifies that the RTM data is the concatenation of:

Encrypted tachograph payload the encryption of the plaintext tachograph payload

DSRC security data described below

The plaintext tachograph payload data format is specified in ~~Appendix-Sub-appendix~~ 1 and further described in ~~Appendix-Sub-appendix~~ 14. This section describes the structure of the DSRC security data; the formal specification is in ~~Appendix-Sub-appendix~~ 1.

CSM_223 The plaintext tachographPayload data communicated by a VU to a Remote Communication Facility (if the RCF is external to the VU) or from the VU to a remote interrogator over the DSRC interface (if the RCF is internal in the VU) shall be protected in encrypt-then-authenticate mode, i.e. the tachograph payload data is encrypted first to ensure message confidentiality, and afterwards a MAC is calculated to ensure data authenticity and integrity.

CSM_224 The DSRC security data shall consist of the concatenation of the following data elements in the following order; see also **Figure 12**:

Current date time TimeReal)	the current date and time of the VU (data type)
Counter	a 3-byte counter, see 0
VU serial number (data type See	the VU's serial number or certificate request ID VuSerialNumber or CertificateRequestID) – CSM_123
DSRC master key version number key from see	the 1-byte version number of the DSRC master which the VU-specific DSRC keys were derived, section 9.2.2.
MAC RTM	the MAC calculated over all previous bytes in the data.

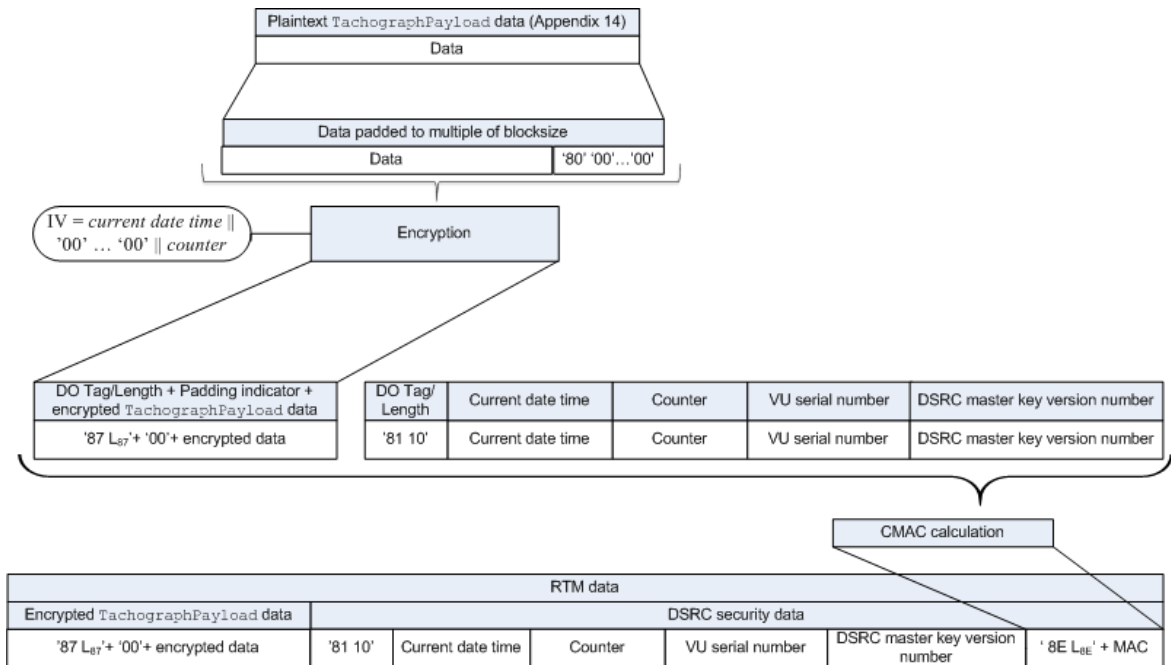
CSM_225 The 3-byte counter in the DSRC security data shall be in MSB-first format. The first time a VU calculates a set of RTM data after it is taken into production, it shall set the value of the counter to 0. The VU shall increase the value of the counter data by 1, each time before it calculates a next set of RTM data.

13.2. Tachograph Payload Encryption and MAC Generation

CSM_226 Given a plaintext data element with data type `TachographPayload` as described in ~~Appendix~~ **Sub-appendix** 14, a VU shall encrypt this data as shown in **Error! Reference source not found.**: the VU's DSRC key for encryption $K_{VU_{DSRC_ENC}}$ (see section ~~9.2.3~~ **9.2.2**) shall be used with AES in the Cipher Block Chaining (CBC) mode of operation, as defined in [ISO 10116], with an interleave parameter $m = 1$. The initialization vector shall be equal to $IV = current\ date\ time \parallel '00\ 00\ 00\ 00\ 00\ 00\ 00\ 00\ 00' \parallel counter$, where *current date time* and *counter* are specified in ~~FCS_366~~ **CSM_224**. The data to be encrypted shall be padded using method 2 defined in [ISO 9797-1].

CSM_227 A VU shall calculate the MAC in the DSRC security data as shown in **Figure 12**: the MAC shall be calculated over all preceding bytes in the RTM data, up to and including the DSRC master key version number, and including the tags and lengths of the data objects. The VU shall use its DSRC key for authenticity $K_{VU_{DSRC_MAC}}$ (see section ~~9.2.3~~ **9.2.2**) with the AES algorithm in CMAC mode as specified in [SP 800-38B]. The length of the MAC shall be linked to the length of the VU-specific DSRC keys, as specified in ~~RCS_192~~ **CSM_50**.

Figure 12
Tachograph payload encryption and M AC generation



13.3. Verification and Decryption of Tachograph Payload

CSM_228 When a remote interrogator receives RTM data from a VU, it shall send the entire RTM data to a control card in the data field of a PROCESS DSRC MESSAGE command, as described in ~~Appendix Sub-appendix 2~~. Then:

1. The control card shall inspect the DSRC master key version number in the DSRC security data. If the control card does not know the indicated DSRC master key, it shall return an error specified in ~~Appendix Sub-appendix 2~~ and abort the process.
2. The control card shall use the indicated DSRC master key in combination with the VU serial number **or the certificate request ID** in the DSRC security data to derive the VU-specific DSRC keys $K_{VU_{DSRC_ENC}}$ and $K_{VU_{DSRC_MAC}}$, as specified in ~~TCS_266~~ **CSM_124**.
3. The control card shall use $K_{VU_{DSRC_MAC}}$ to verify the MAC in the DSRC security data, as specified in ~~TCS_369~~ **CSM_227**. If the MAC is incorrect, the control card shall return an error specified in ~~Appendix Sub-appendix 2~~ and abort the process.
4. The control card shall use $K_{VU_{DSRC_ENC}}$ to decrypt the encrypted tachograph payload, as specified in ~~TCS_368~~ **CSM_226**. The control card shall remove the padding and shall return the decrypted tachograph payload data to the remote interrogator.

CSM_229 In order to prevent replay attacks, the remote interrogator shall verify the freshness of the RTM data by verifying that the *current date time* in the DSRC security data does not deviate too much from the current time of the remote interrogator.

Notes:

- This requires the remote interrogator to have an accurate and reliable source of time.
- Since ~~Appendix~~ **Sub-appendix** 14 requires a VU to calculate a new set of RTM data every 60 seconds, and the clock of the VU is allowed to deviate 1 minute from the real time, a lower limit for the freshness of the RTM data is 2 minutes. The actual freshness to be required also depends on the accuracy of the clock of the remote interrogator.

CSM_230 When a workshop verifies the correct functioning of the DSRC functionality of a VU, it shall send the entire RTM data received from the VU to a workshop card in the data field of a PROCESS DSRC MESSAGE command, as described in ~~Appendix~~ **Sub-appendix** 2. The workshop card shall perform all checks and actions specified in ~~TCS_370~~ CSM_228.

14. Signing Data Downloads and Verifying Signatures

14.1. General

CSM_231 The Intelligent Dedicated Equipment (IDE) shall store data received from a VU or a card during one download session within one physical data file. Data may be stored on an ~~ESM~~ (external storage medium). This file contains digital signatures over data blocks, as specified in ~~Appendix~~ **Sub-appendix** 7. This file shall also contain the following certificates (refer to section 9.1):

- In case of a VU download:
 - The VU_Sign certificate
 - The MSCA_VU-EGF certificate containing the public key to be used for verification of the VU_Sign certificate
- In case of a Card download:
 - The Card_Sign certificate
 - The MSCA_Card certificate containing the public key to be used for verification of the Card_Sign certificate

CSM_232 The IDE shall also dispose of.

- In case it uses a control card to verify the signature, as shown in **Figure 13**: The link certificate linking the latest EUR certificate to the EUR certificate whose validity period directly precedes it, if existing.
- In case it verifies the signature itself: all valid European root certificates.

Note: the method the IDE uses to retrieve these certificates is not specified in this ~~Appendix~~ **sub-appendix**.

14.2. Signature generation

CSM_233 The signing algorithm to create digital signatures over downloaded data shall be ECDSA as specified in [DSS], using the hashing algorithm linked to the key size of the VU or the card, as specified in ~~TCS_192~~ CSM_50. The signature format shall be plain, as specified in [TR-03111].

14.3. Signature verification

CSM_234 An IDE may perform verification of a signature over downloaded data itself or it may use a control card for this purpose. In case it uses a control card, signature verification shall take place as shown in **Figure 13**. **For verifying the temporal validity of a certificate presented by the IDE, the control card shall use its internal current time, as specified in CSM_167. The control card shall update its current time if the Effective Date of an authentic ‘valid source of time’ certificate is more recent than the card’s current time. The card shall accept only the following certificates as a valid source of time:**

- **Second-generation ERCA link certificates**
- **Second-generation MSCA certificates**
- **Second-generation VU_Sign or Card_Sign certificates issued by the same country as the control card’s own card certificate.**

In case it performs signature verification itself, the IDE shall verify the authenticity and validity of all certificates in the certificate chain in the data file, and it shall verify the signature over the data following the signature scheme defined in [DSS]. **In both cases, for every certificate read from the data file, it is necessary to verify that the Certificate Holder Authorisation (CHA) field is correct:**

- **The CHA field of the EQT certificate shall indicate a VU or Card (as applicable) certificate for signing (see Sub-appendix 1, data type EquipmentType).**
- **The CHA of the EQT.CA certificate shall indicate an MSCA.**
- **The CHA of the EQT.Link certificate shall indicate the ERCA.**

Notes to **Figure 13** :

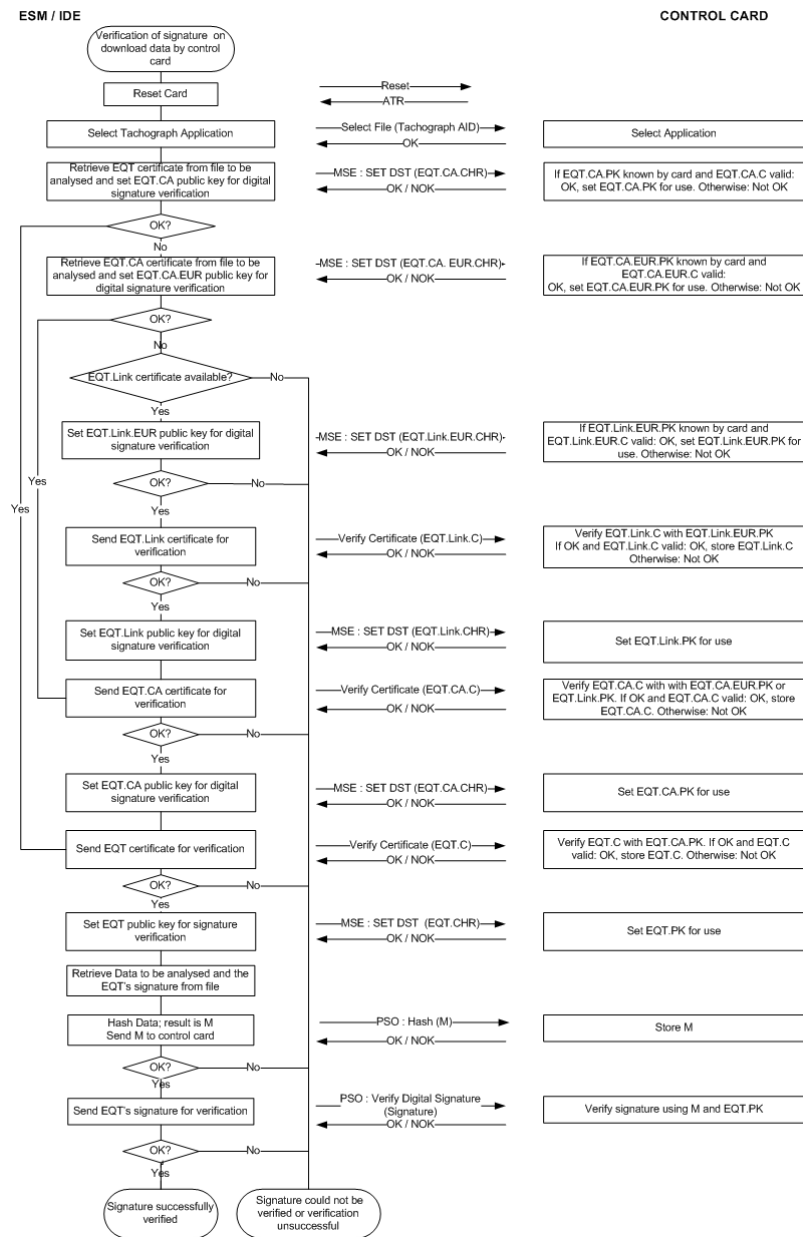
- The equipment that signed the data to be analysed is denoted EQT.
- The EQT certificates and public keys mentioned in the figure are those for signing, i.e. VU_Sign or Card_Sign.
- The EQT.CA certificates and public keys mentioned in the figure are those for signing VU or Card certificates, as applicable.
- The EQT.CA.EUR certificate mentioned in the figure is the European root certificate that is indicated in the CAR of the EQT.CA certificate.
- The EQT.Link certificate mentioned in the figure is the EQT’s link certificate, if present. As specified in section 9.1.2, this is a link certificate for a new ~~European~~ root key pair created by the ERCA and signed with the previous ~~European~~ root private key.
- The EQT.Link.EUR certificate is the ~~European~~-root certificate that is indicated in the CAR of the EQT.Link certificate.

CSM_235 For calculating the hash M sent to the control card in the PSO:Hash command, the IDE shall use the hashing algorithm linked to the key size of the VU or the card from which the data is downloaded, as specified in ~~TCS_192~~ **CSM_50**.

CSM_236 For verifying the EQT’s signature, the control card shall follow the signature scheme defined in [DSS].

Note: This document does not specify any action to undertake if a signature over a downloaded data file cannot be verified or if the verification is unsuccessful.

Figure 13
Protocol for verification of the signature over a downloaded data file



Sub-appendix 12. Positioning based on Global Navigation Satellite System (GNSS)

TABLE OF CONTENT

APPENDIX 12 POSITIONING BASED ON GLOBAL NAVIGATION SATELLITE SYSTEM (GNSS)

Introduction.....	491
1.1. Scope.....	491
1.1.1 References	491
1.2. Acronyms and notations	492
2. SPECIFICATION-BASIC CHARACTERISTICS OF THE GNSS RECEIVER	492
3. NMEA-SENTENCES PROVIDED BY THE GNSS RECEIVER.....	492
4. VEHICLE UNIT WITH AN EXTERNAL GNSS FACILITY	494
4.1. Configuration.....	497
4.1.1 Main components and interfaces	497
4.1.2 External GNSS facility state at the end of production	497
4.2. Communication between the external GNSS facility and the vehicle unit	497
4.2.1 Communication Protocol	498
4.2.2 Secure transfer of GNSS data.....	498
4.2.3 Structure of the Read Record command	500
4.2.4 Structure of the WriteRecord command	501
4.2.5 Other commands	502
4.3. Coupling, mutual authentication and session key agreement of the external GNSS facility with vehicle unit.....	503
4.4. Error Handling.....	503
4.4.1 Communication error with the external GNSS facility.....	503
4.4.2 Breach of the physical integrity of the external GNSS facility	504
4.4.3 Absence of position information from GNSS receiver.....	504
4.4.4 External GNSS facility certificate expired.....	504
5. VEHICLE UNIT WITHOUT AN EXTERNAL GNSS FACILITY	505
5.1. Configuration.....	505
5.2. Error HandlingTransfer of information from the GNSS Receiver to the VU	505
5.2.1 Absence of position information from GNSS receiver.....	505
5.3. TRANSFER OF INFORMATION FROM THE VU TO THE GNSS RECEIVER	505
5.4. ERROR HANDLING	505
5.4.1 Absence of position information from GNSS receiver.....	505
6. GNSS TIME CONFLICT POSITION DATA PROCESSING AND RECORDING BY THE VU	505
7. GNSS TIME CONFLICT	507
8. VEHICLE MOTION CONFLICT	507

1. Introduction

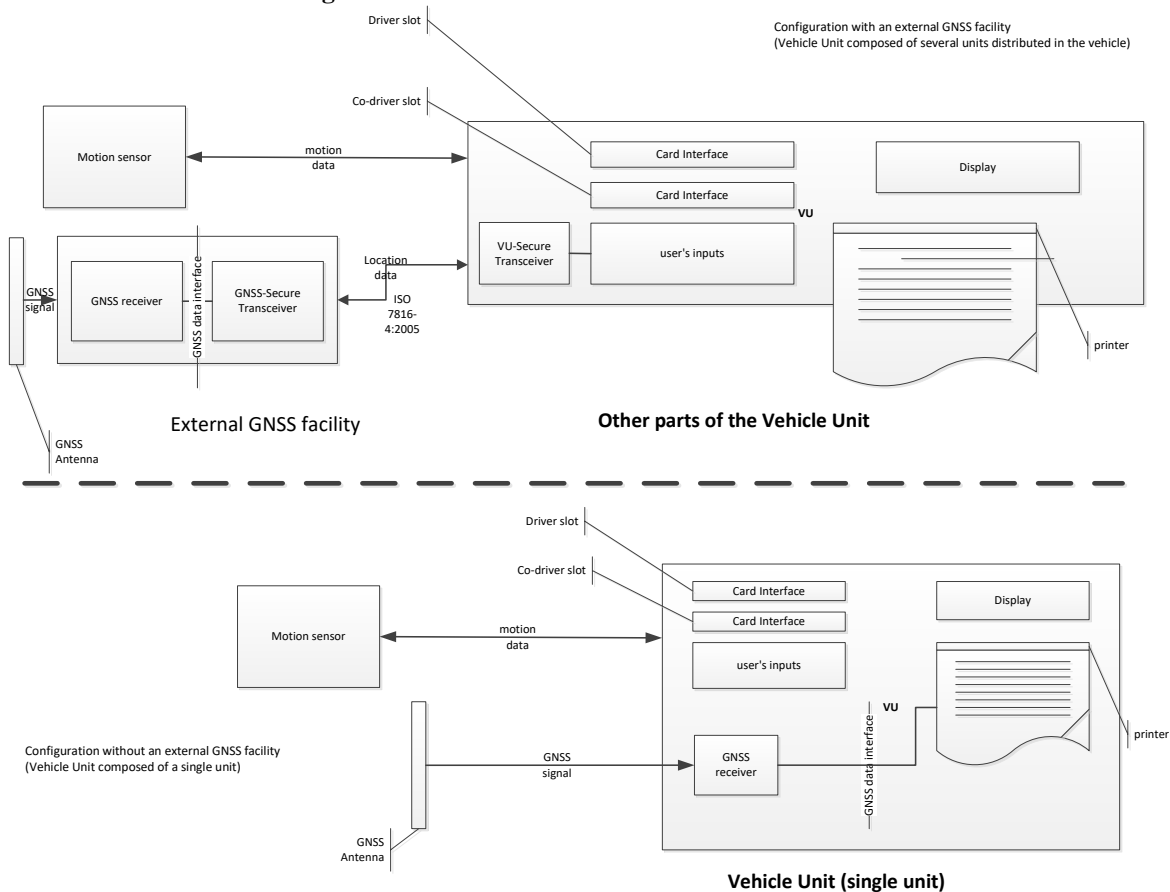
This ~~Appendix~~ **sub-appendix** provides the technical requirements for the GNSS receiver and GNSS data used by the Vehicle Unit, including the protocols that must be implemented to assure the secure and correct data transfer of the positioning information. ~~The main articles in the Regulation (EU) No. 165/2014 driving these requirements are ‘Article 8 Recording of the position of the vehicle at certain points during the daily working period,’ ‘Article 10 Interface with Intelligent Transport Systems’ and Article 11 Detailed provisions for smart tachograph.’~~

1.1. Scope

GNS_1 The Vehicle Unit shall collect location data from at least one GNSS satellite network ~~to support the implementation of Article 8.~~

The Vehicle Unit may be with or without an external GNSS ~~to support the implementation of article 8~~ facility as described in Figure 144:

Figure 1
Different configurations for GNSS receiver.



1.1.1 References

The following references are used in this part of this sub-appendix.

NMEA NMEA (National Marine Electronics Association) 0183 Interface Standard, V4.11

1.2. Acronyms and notations

The following acronyms are used in this **sub**-appendix:

DOP	Dilution of Precision
EGF	Elementary file GNSS Facility
EGNOS	European Geostationary Navigation Overlay Service
GNSS	Global Navigation Satellite System
GSA	GPS DOP and active satellites
HDOP	Horizontal Dilution of Precision
ICD	Interface Control Document
NMEA	National Marine Electronics Association
PDOP	Position Dilution of Precision
RMC	Recommended Minimum Specific
SIS	Signal in Space
VDOP	Vertical Dilution of Precision
VU	Vehicle Unit
OSNMA	Galileo Open Service Navigation Messages Authentication
RTC	Real Time Clock

2. ~~Specification~~ Basic characteristics of the GNSS receiver

~~Regardless of the configuration of the Smart Tachograph with or without an external GNSS facility, the provision of accurate and reliable positioning formation is an essential element of the effective operation of the Smart Tachograph. Therefore, it is appropriate to require its compatibility with the services provided by the Galileo and European Geostationary Navigation Overlay Services (EGNOS) programme as set out in Regulation (EU) No. 1285/2013 of the European Parliament and the Council¹⁴. The system established under the Galileo programme is an independent global satellite navigation system and the one established under the EGNOS programme is a regional satellite navigation system improving the quality of the Global Positioning System Signal.~~

GNS_2 Manufacturers shall ensure that the GNSS receivers in the Smart Tachographs are compatible with the positioning services provided by ~~the~~ **GPS, GLONASS and Galileo** ~~and the EGNOS systems~~. Manufacturers may also choose, in addition, compatibility with other satellite navigation systems.

GNS_3 The GNSS receiver shall have the capability to support **Navigation Messages Authentication** on the Open Service of Galileo (**OSNMA**) ~~when such service will be provided~~

¹⁴ ~~Regulation (EU) No. 128 of the European Parliament and the Council of 11 December 2013 on the implementation and exploitation of European satellite navigation systems and repealing Council Regulation (EC) 876/2002 and Regulation (EC) no 683/2008 of the European Parliament of the Council (OJL 347,2013 p.1).~~

~~by the Galileo system and supported by GNSS receiver manufacturers. However, for smart tachographs introduced in the market before the previous conditions are satisfied and not having the capability to support Authentication of the Open Service of Galileo, no retrofitting will be required.~~

GNS_3a The GNSS receiver shall perform a number of consistency checks in order to verify that the measurements computed by the GNSS receiver on the basis of the OSNMA data have resulted in the correct information about the position, velocity and data of the vehicle, and have therefore not been influenced by any external attack such as meaconing. These consistency checks shall consist, for instance, of:

- detection of abnormal power emissions by means of combined monitoring of the Automatic Gain Control (AGC) and Carrier-to-Noise density ratio (C/N0),
- pseudorange measurement consistency and Doppler measurement consistency over time, including the detection of abrupt measurement jumps,
- receiver autonomous integrity monitoring (RAIM) techniques, including the detection of inconsistent measurements with the estimated position,
- position and velocity checks, including abnormal position and velocity solutions, sudden jumps and behaviour not consistent with the dynamics of the vehicle,
- time and frequency consistency, including clock jumps and drifts that are not consistent with the receiver clock characteristics.

GNS_3b The European Commission shall develop and approve the following documents:

- A Signal in Space Interface Control Document (SIS ICD), specifying in detail the OSNMA information transmitted in the Galileo signal.
- OSNMA Receiver Guidelines, providing the requirements and processes in the receivers to guarantee a secure implementation of OSNMA, as well as recommendations to enhance OSNMA performance.

GNSS receivers fitted in tachographs, either internal or external, shall be constructed in accordance with the SIS ICD and the OSNMA receiver guidelines.

GNS_3c The GNSS receiver shall provide position messages, called authenticated position messages, which are elaborated using only satellites from which the authenticity of the navigation messages has been successfully verified.

GNS_3d The GNSS receiver shall also provide standard position messages, elaborated using the satellites in view, regardless whether they are authenticated or not.

GNS_3e The GNSS receiver shall use the VU Real Time Clock (RTC) as time reference for the loose time synchronisation necessary for OSNMA.

GNS3f The VU RTC time shall be provided to the GNSS receiver by the VU.

GNS3f The maximal time drift specified in requirement 41 of Appendix 1C, shall be provided to the GNSS receiver by the VU, along with the VU RTC time.

~~GNSS receivers may be also capable of receiving and processing SBAS signals.~~

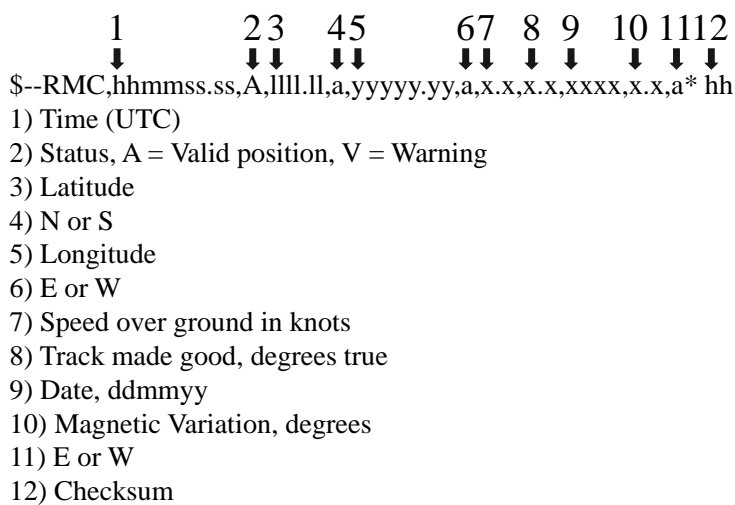
3. NMEA Sentences provided by the GNSS receiver

This section describes the ~~NMEA~~ sentences used in the functioning of the Smart Tachograph, **for transmitting standard and authenticated position messages**. This section is valid both for the configuration of the Smart Tachograph with or without an external GNSS facility.

GNS_4 The **standard position**~~location~~ data is based on the NMEA sentence Recommended Minimum Specific (RMC) GNSS Data, which contains the Position information (Latitude, Longitude), Time in UTC format (~~hhmmss~~**hhmmss.ss** ~~Contracting Parties~~), and Speed Over Ground in Knots plus additional values.

The format of the RMC sentence is the following (as from NMEA V4.1 standard):

Figure 2
Structure of the RMC sentence



The Status gives indication if the GNSS signal is available. Until the value of the Status is not set to A, the received data (e.g., on Time or Latitude/Longitude) cannot be used to record the position of the vehicle in the VU.

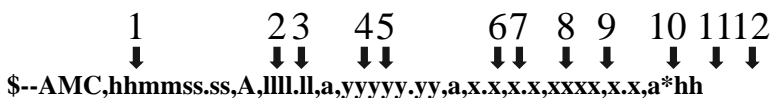
The resolution of the position is based on the format of the RMC sentence described above. The first part of the fields 3) and 5) (~~the first two numbers~~) are used to represent the degrees. The rest are used to represent the minutes with three decimals. So the resolution is 1/1000 of minute or 1/60000 of degree (because one minute is 1/60 of a degree).

GNS_4a The **authenticated position data** is based on a NMEA-like sentence, **Authenticated Minimum Specific (AMC) Data, which contains the Position information (Latitude, Longitude), Time in UTC format (hhmmss.ss), and Speed Over Ground in Knots plus additional values.**

The format of the AMC sentence is the following (as from NMEA V4.11 standard, except for value number 2):

Figure 3

Structure of the AMC sentence



- 1) Time (UTC)
- 2) Status, A=Authenticated position (established using at least 4 satellites from which the authenticity of the navigation messages has been successfully verified), J=jamming or O=other GNSS attack in the absence of failed authentication of navigation messages (by implemented consistency checks according to GNS_3a), F=failed authentication of navigation messages (as detected by OSNMA verifications specified in the documents referred to in GNS_3b), V=Void (authenticated position is not available for any other reason)
- 3) Latitude
- 4) N or S
- 5) Longitude
- 6) E or W
- 7) Speed over ground in knots
- 8) Track made good, degrees true
- 9) Date, ddmmyy
- 10) Magnetic Variation, degrees
- 11) E or W
- 12) Checksum

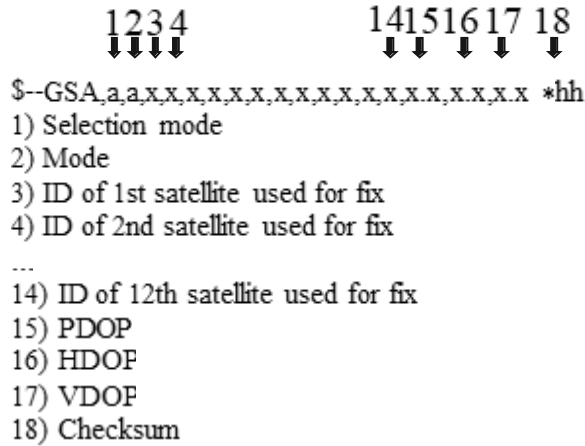
The Status gives indication if an authenticated GNSS position is available, if an attack on the GNSS signals has been detected, if authentication of navigation messages has failed, or if GNSS position is void. When the value of the Status is not set to 'A', the received data (e.g. Time or Latitude/Longitude) are considered to be not valid, and may not be used to record the position of the vehicle in the VU. When the value of the Status is set to 'J' (jamming), 'O' (other GNSS attack), or 'F' (failed authentication of navigation messages), a GNSS anomaly event shall be recorded in the VU, as defined in Appendix 1C and Sub-appendix 1 (EventFaultCode).

GNS_5 The Vehicle Unit shall store in the VU database the position information for latitude and longitude with a resolution of 1/10 of minute or 1/600 of a degree as described in ~~Appendix~~ **Sub-appendix 1** for type GeoCoordinates.

The GPS DOP and active satellites (GSA) command, **as from NMEA V4.11 standard**, can be used by the VU to determine and record the signal availability and accuracy **of standard positions**. In particular the HDOP is used to provide an indication on the level of accuracy of the recorded location data (see 4.2.2). The VU will store the value of the Horizontal Dilution of Precision (HDOP) calculated as the minimum of the HDOP values collected on the available GNSS systems.

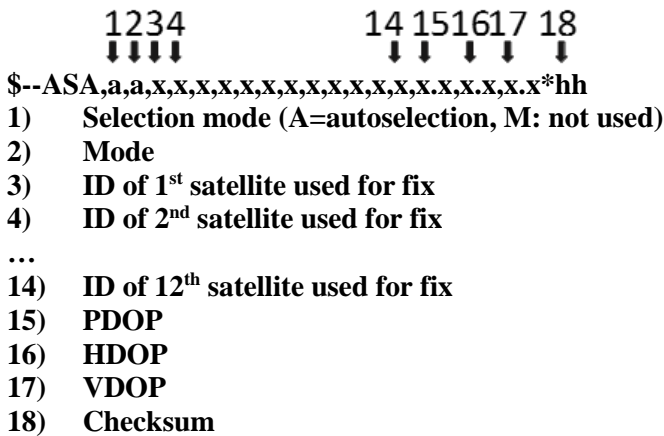
The GNSS System Id indicates ~~GPS, Glonass, Beidou or~~ **the corresponding NMEA Id. for every GNSS constellation and** Satellite-Based Augmentation System (SBAS).

Figure 43
Structure of the GSA sentence (standard positions)



Similarly, the NMEA-like sentence authenticated active satellites (ASA) command can be used by the VU to determine and record the signal availability and accuracy of authenticated positions. Values 1 to 18 are defined in NMEA V4.11 standard.

Figure 5
Structure of the ASA sentence (authenticated positions)



Where the Mode (2) gives an indication if a fix is not available (Mode 1) or a fix is available for 2D (Mode 2) or 3D (Mode 3).

GNS_6 When an external GNSS facility is used, the GSA sentence shall be stored in the GNSS Secure Transceiver with record number '02' to '06', and the ASA sentence shall be stored with record number '12' to '16'.

GNS_7 The maximum size of the NMEA-sentences (e.g., RMC, AMC, GSA, ASA or others), which can be used for the sizing of the read record command shall be 85 bytes (see ~~Table 49~~ Table 1).

4. Vehicle Unit with an external GNSS Facility

4.1. Configuration

4.1.1 Main components and interfaces

In this configuration, the GNSS receiver is a part of the external GNSS facility.

GNS_8 The external GNSS facility must be powered with a specific vehicle interface.

GNS_9 The external GNSS facility shall consist of the following components (see Figure 64 47):

(a) A commercial GNSS receiver to provide the position data through the GNSS data interface. For example, the GNSS data interface can be NMEA standard V4.110 where the GNSS receiver acts as a talker and transmits NMEA sentences to the GNSS Secure Transceiver with a frequency of 1Hz for the pre-defined set of NMEA sentences, which must include at least the RMC, AMC, and GSA and ASA sentences. The implementation of the GNSS data interface is a choice of the manufacturers of the external GNSS facility.

(b) A transceiver unit (GNSS Secure Transceiver) with the capability to support standard ISO/IEC 7816-4:2013 (see ~~5.2.1~~ 4.2.1) to communicate with the vehicle unit and support the GNSS data interface to the GNSS receiver. The unit is provided with a memory to store the identification data of the GNSS receiver and external GNSS facility.

(c) An enclosure system with tamper detection function, which encapsulates both the GNSS receiver and the GNSS Secure Transceiver. The tamper detection function shall implement the security protection measures as requested in the Protection Profile of the Smart Tachograph.

(d) A GNSS antenna installed on the vehicle and connected to the GNSS receiver through the enclosure system.

GNS_10 The external GNSS facility has at least the following external interfaces:

(a) the interface to the GNSS antenna installed on the vehicle truck, if an external antenna is used.

(b) the interface to the Vehicle Unit.

GNS_11 In the VU, the VU Secure Transceiver is the other end of the secure communication with the GNSS Secure Transceiver and it must support ISO/IEC 7816-4:2013 for the connection to the external GNSS facility.

GNS_12 For the physical layer of the communication with the external GNSS facility, the vehicle unit shall support ISO/IEC 7816-12:2005 or another standard able to support ISO/IEC 7816-4:2013. (see ~~5.2.1~~ 4.2.1).

4.1.2 External GNSS facility state at the end of production

GNS_13 The external GNSS facility shall store the following values in the non-volatile memory of the GNSS Secure Transceiver when it leaves the factory:

- the EGF_MA key pair and corresponding certificate,
- the MSCA_VU-EGF certificate containing the MSCA_VU-EGF.PK public key to be used for verification of the EGF_MA certificate,
- the EUR certificate containing the EUR.PK public key to be used for verification of the MSCA_VU-EGF certificate,

- the EUR certificate whose validity period directly precedes the validity period of the EUR certificate to be used to verify the MSCA_VU-EGF certificate, if existing,
- the link certificate linking these two EUR certificates, if existing,
- the extended serial-number of the external GNSS facility,
- operating system identifier of the GNSS facility,
- type approval number of the external GNSS facility;
- Identifier of the security component of the external GNSS module.

4.2. Communication between the external GNSS facility and the vehicle unit

4.2.1 Communication Protocol

GNS_14 The communication protocol between the external GNSS facility and the vehicle unit shall support three functions:

1. The collection and distribution of GNSS data (e.g., position, timing, speed),
2. The collection of the configuration data of the external GNSS facility,
3. The management protocol to support the coupling, mutual authentication and session key agreement between the external GNSS facility and the VU.

4. The transmission to the external GNSS facility of the VU RTC time and of the maximal difference between true time and the VU RTC time.

GNS_15 The communication protocol shall be based on standard ISO/IEC 7816-4:2013 with the VU Secure Transceiver playing the master role and the GNSS Secure Transceiver playing the slave role. The physical connection between the external GNSS facility and the vehicle unit is based on ISO/IEC 7816-12:2005 or another standard able to support ISO/IEC 7816-4:2013.

GNS_16 In the communication protocol, extended length fields shall not supported.

GNS_17 The communication protocol of ISO 7816 (both *-4:2013 and *-12:2005) between the external GNSS facility and the VU shall be set to T=1.

GNS_18 Regarding the functions 1) the collection and distribution of GNSS data and 2) the collection of the configuration data of the external GNSS facility and 3) management protocol, the GNSS Secure Transceiver shall simulate a smart card with a file system architecture composed by a Master File (MF), a ~~Directory~~ **Dedicated** File (DF) with Application Identifier specified in ~~Appendix~~ **Sub-appendix** 1 chapter 6.2 (' FF 44 54 45 47 4D') and with 3 EFs containing certificates and one single Elementary File (EF.EGF) with file identifier equal to '2F2F' as described in ~~Table 49~~ **Table 1**.

GNS_18a Regarding the function 4) the transmission to the external GNSS facility of the VU RTC time and of the maximal difference between true time and the VU RTC time, the GNSS Secure Transceiver shall use an EF (EF VU) in the same DF with file identifier equal to '2F30' as described in Table 1.

GNS_19 The GNSS Secure Transceiver shall store the data coming from the GNSS receiver and the configuration in the EF.EGF. This is a linear, variable-length record file with an identifier equal to '2F2F' in hexadecimal format.

GNS_19a The GNSS Secure Transceiver shall store the data coming from the VU in the EF VU. This is a linear, fixed-length record file with an identifier equal to '2F30' in hexadecimal format.

GNS_20 The GNSS Secure Transceiver shall use a memory to store the data **and be** able to perform **as many at least 20 millions write/read/write cycles as needed during a lifetime of at least 15 years**. Apart from this aspect, the internal design and implementation of the GNSS Secure Transceiver is left to the manufacturers.

The mapping of record numbers and data is provided in ~~Table 49~~ **Table 1**. Note that there are ~~four~~ five GSA sentences for the ~~four satellite systems~~ **GNSS constellations** and Satellite-Based Augmentation System (SBAS).

GNS_21 The file structure is provided in ~~Table 49~~ **Table 1**. For the access conditions (ALW, NEV, SM-MAC) see ~~Appendix Sub-appendix 2~~ chapter 3.5.

File	File ID	Access conditions		
		Read	Update	Encrypted
MF	3F00			
EF.ICC	0002	ALW	NEV (by VU)	No
DF GNSS Facility	0501	ALW	NEV	No
EF EGF_MACertificate	C100	ALW	NEV	No
EF CA_Certificate	C108	ALW	NEV	No
EF Link_Certificate	C109	ALW	NEV	No
EF -EGF	2F2F	SM-MAC	NEV (by VU)	No
EF VU	2F30	SM-MAC	SM-MAC	No

File / Data element	Record no	Size (bytes)		Default values
		Min	Max	
MF		552	1031	
EF.ICC				
sensorGNSSSerialNumber		8	8	
DF GNSS Facility		612	1023	
EF EGF_MACertificate		204	341	
EGFCertificate		204	341	{00..00}
EF CA_Certificate		204	341	
MemberStateCertificate		204	341	{00..00}
EF Link_Certificate		204	341	
LinkCertificate		204	341	{00..00}
EF -EGF				
RMC NMEA Sentence	'01'	85	85	
1st GSA NMEA Sentence	'02'	85	85	
2nd GSA NMEA Sentence	'03'	85	85	

<i>File / Data element</i>	<i>Record no</i>	<i>Size (bytes)</i>		<i>Default values</i>
3rd GSA NMEA Sentence	'04'	85	85	
4th GSA NMEA Sentence	'05'	85	85	
5th GSA NMEA Sentence	'06'	85	85	
Extended serial-number of the external GNSS facility defined in Appendix Sub-appendix 1 as SensorGNSSSerialNumber.	'07'	8	8	
Operating system identifier of the GNSS Secure Transceiver defined in Appendix Sub-appendix 1 as SensorOSIdentifier.	'08'	2	2	
Type approval number of the external GNSS facility defined in Appendix Sub-appendix 1 as SensorExternalGNSSApprovalNumber.	'09'	16	16	
Identifier of the security component of the external GNSS facility defined in Appendix Sub-appendix 1 as SensorExternalGNSSSCIdentifier	'10'	8	8	
AMC Sentence	'11'	85	85	
1st ASA Sentence	'12'	85	85	
2nd ASA Sentence	'13'	85	85	
3rd ASA Sentence	'14'	85	85	
4th ASA Sentence	'15'	85	85	
5th ASA Sentence	'16'	85	85	
RFU – Reserved for Future Use	From '17+' to 'FD'			
EF VU				
VuRtcTime (see Sub-appendix 1)	'01'	4	4	{00..00}
VuGnssMaximalTimeDifference (see Sub-appendix 1)	'02'	2	2	{00..00}

Table 1
File Structure

4.2.2 Secure transfer of GNSS data

GNS_22 The secure transfer of GNSS position data, **VU RTC time and maximal time difference between true time and the VU RTC time** shall be allowed only in the following conditions:

1. The coupling process has been completed as described in ~~Appendix~~ **Sub-appendix 11**. Common security mechanisms.
2. The periodic mutual authentication and session key agreement between the VU and the external GNSS facility also described in ~~Appendix~~ **Sub-appendix 11**. Common security mechanisms has been executed with the indicated frequency.

GNS_23 Every T seconds, where T is a value lower or equal to ~~120~~, unless coupling or mutual authentication and session key agreement takes place, the VU requests from the external GNSS facility the position information on the basis of the following flow:

1. The VU requests location data from the External GNSS facility together with Dilution of Precision data (from the GSA and ASA NMEA sentences). The VU Secure Transceiver shall use the ISO/IEC 7816-4:2013 SELECT and READ RECORD(S) commands in secure messaging authentication-only mode as described in ~~Appendix Sub-appendix 11~~ **section 11.5 of Sub-appendix 11** with the file identifier “2F2F” and RECORD number equal to “01” for RMC NMEA sentence, and ‘02’, ‘03’, ‘04’, ‘05’, ‘06’ for GSA NMEA sentence, **‘11’ for AMC sentence, and ‘12’, ‘13’, ‘14’, ‘15’, ‘16’ for ASA sentence.**
2. The last ~~position~~ **location** data received is stored in the EF with identifier ‘2F2F’ and the records described in ~~Table 49~~ **Table 1** in the GNSS secure transceiver as the GNSS secure transceiver receives NMEA data with a frequency of at least 1 Hz from the GNSS receiver through the GNSS data interface.
3. The GNSS Secure Transceiver sends the response to the VU Secure Transceiver by using the APDU response message in secure messaging authentication-only mode as described in ~~Appendix Sub-appendix 11~~ **section 11.5 of Sub-appendix 11.**
4. The VU Secure Transceiver checks the authenticity and integrity of the received response. In case of positive outcome, the ~~location~~ **position** data is transferred to the VU processor through the GNSS data interface.
5. The VU processor checks the received data extracting the information (e.g., latitude, longitude, time) from the RMC NMEA sentence. The RMC NMEA sentence includes the information if the **non-authenticated** position is valid. If the **non-authenticated** position is not valid, the location data is not available yet and it cannot be used to record the position of the vehicle. If the position is valid, the VU processor also extracts the values of HDOP from GSA NMEA sentences and calculates the ~~average~~ **minimum** value on the available satellite systems (i.e., when the fix is available).
6. The VU processor **also extracts the information (e.g., latitude, longitude, time) from the AMC sentence. The AMC sentence includes the information if the trusted position is not valid or GNSS signal has been attacked. If the position is valid, the VU processor also extracts the values of HDOP from ASA sentences and calculates the minimum value on the available satellite systems (i.e., when the fix is available)** stores the received and processed information such as latitude, longitude, time and speed in the VU in the format defined in ~~Appendix Sub-appendix 1 Data Dictionary as GeoCoordinates~~ together with the value of HDOP calculated as the minimum of the HDOP values collected on the available GNSS systems.

GNS_23a The VU shall also write VU RTC time and maximal time difference between true time and the VU RTC time as needed, by using the ISO/IEC 7816-4:2013 SELECT and WRITE RECORD(S) commands in secure messaging authentication-only mode as described in section 11.5 of Sub-appendix 11 with the file identifier ‘2F30’ and RECORD number equal to ‘01’ for VuRtcTime and ‘02’ for MaximalTimeDifference.

4.2.3 Structure of the Read Record command

This section describes in detail the structure of the Read Record command. Secure messaging (authentication-only mode) is added as described in ~~Appendix Sub-appendix 11~~ Common security mechanisms.

GNS_24 The command shall support the Secure Messaging authentication-only-mode, see ~~Appendix Sub-appendix 11~~.

GNS_25 Command Message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure messaging asked.
INS	1	'B2h'	Read Record
P1	1	'XXh'	Record number ('00' references the current record)
P2	1	'04h'	Read the record with the record number indicated in P1
Le	1	'XXh'	Length of data expected. Number of Bytes to be read.

GNS_26 The record referenced in P1 becomes the current record.

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

- If the command is successful, the GNSS secure transceiver returns '9000'.
- If the current file is not record oriented, the GNSS secure transceiver returns '6981'.
- If the command is used with P1 = '00' but there is no current EF the GNSS secure transceiver returns '6986' (command not allowed).
- If the record is not found, the GNSS secure transceiver returns '6A83'.
- If the external GNSS facility has detected tampering, it shall return status words '6690'.

~~GNS_27 The GNSS Secure Transceiver shall support the following tachograph generation 2 commands specified in Appendix Sub-appendix 2:~~

Command	Reference
Select	Appendix Sub-appendix 2 chapter 3.5.1
Read Binary	Appendix Sub-appendix 2 chapter 3.5.2
Get Challenge	Appendix Sub-appendix 2 chapter 3.5.4
PSO: Verify Certificate	Appendix Sub-appendix 2 chapter 3.5.7
External Authenticate	Appendix Sub-appendix 2 chapter 3.5.9
General Authenticate	Appendix Sub-appendix 2 chapter 3.5.10
MSE:SET	Appendix Sub-appendix 2 chapter 3.5.11

4.2.4 Structure of the Read Record command

This section describes in detail the structure of the Write Record command. Secure messaging (authentication-only mode) is added as described in Sub-appendix 11 Common security mechanisms.

GNS_26a The command shall support the Secure Messaging authentication-only-mode, see Sub-appendix 11.

GNS_26b Command Message

Byte	Length	Value	Description
CLA	1	'0Ch'	Secure messaging asked.
INS	1	'D2h'	Write Record
P1	1	'XXh'	Record number ('00' references the current record)

P2	1	'04h'	Write the record with the record number indicated in P1
Data	X	'XXh'	Data

GNS_26c The record referenced in P1 becomes the current record.

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

- If the command is successful, the GNSS secure transceiver returns '9000'.
- If the current file is not record oriented, the GNSS secure transceiver returns '6981'.
- If the command is used with P1 = '00' but there is no current EF the GNSS secure transceiver returns '6986' (command not allowed).
- If the record is not found, the GNSS secure transceiver returns '6A83'.
- If the external GNSS facility has detected tampering, it shall return status words '6690'.

4.2.5 Other commands

GNS_27 The GNSS Secure Transceiver shall support the following tachograph generation 2 commands specified in Sub-appendix 2:

Command	Reference
Select	Sub-appendix 2 chapter 3.5.1
Read Binary	Sub-appendix 2 chapter 3.5.2
Get Challenge	Sub-appendix 2 chapter 3.5.4
PSO: Verify Certificate	Sub-appendix 2 chapter 3.5.7
External Authenticate	Sub-appendix 2 chapter 3.5.9
General Authenticate	Sub-appendix 2 chapter 3.5.10
MSE:SET	Sub-appendix 2 chapter 3.5.11

4.3. Coupling, mutual authentication and session key agreement of the external GNSS facility with vehicle unit

The coupling, mutual authentication and session key agreement of the external GNSS facility with the vehicle unit is described in ~~Appendix~~ **Sub-appendix** 11. Common security mechanisms, Chapter 11.

4.4. Error Handling

This section describes how potential error conditions by the external GNSS facility are addressed and recorded in the VU.

4.4.1 Communication error with the external GNSS facility

GNS_28 If the VU does not manage to communicate to the coupled external GNSS facility for more than 20 continuous minutes, the VU shall generate and record in the VU an event of type EventFaultType with the value of enum 53'H External '0E'H **Communication error with the external GNSS facility** and with the timestamp set to the current time. The event will be generated only if the following two conditions are satisfied: a) the Smart Tachograph is not in calibration mode and b) the vehicle is moving. In this context, a communication error is triggered when the VU Secure Transceiver does not receive a response message after a request message as described in ~~5.2~~ **4.2**.

4.4.2 Breach of the physical integrity of the external GNSS facility

GNS_29 If the external GNSS facility has been breached, the GNSS Secure Transceiver shall **ensure that cryptographic material is unavailable**~~erase all its memory including cryptographic material~~. As described in GNS_25 and GNS_26, the VU shall detect tampering if the Response has status ‘6690’. The VU shall then generate an event **as defined in requirement 85 of Appendix 1C and Sub-appendix 1** of type (EventFaultType for enum ~~‘55’H ‘19’H~~ Tamper detection of GNSS. **Alternately, the external GNSS facility may not respond to VU requests without secure messaging and with status ‘6A88’**~~any external request anymore~~.

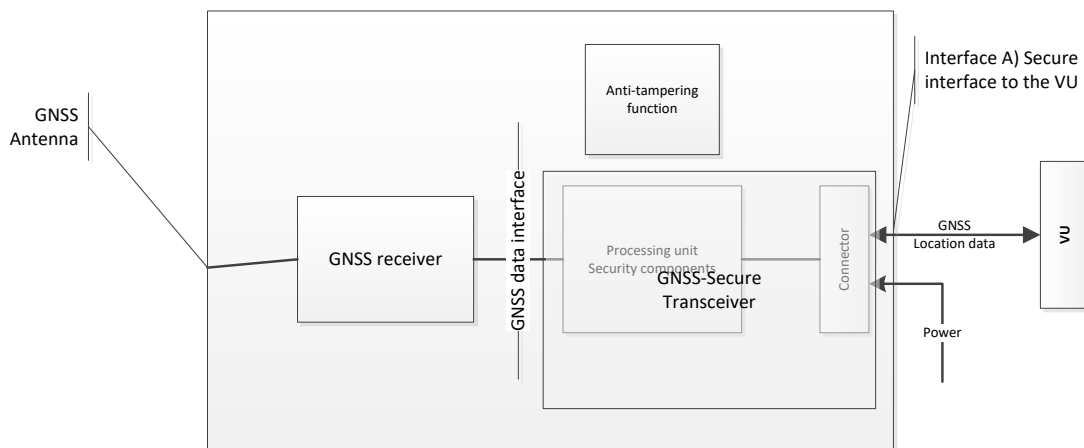
4.4.3 Absence of position information from GNSS receiver

GNS_30 If the GNSS Secure Transceiver does not receive data from the GNSS receiver ~~for more than 3 continuous hours~~, the GNSS Secure Transceiver shall generate a response message to the READ RECORD command with RECORD number equal to ‘01’ with a Data Field of 12 bytes all set to 0xFF. Upon reception of the Response message with this value of the data field, the VU shall generate and record an ~~event of type EventFaultType enum ‘52’H external ‘0D’H~~ **Absence of position information from GNSS receiver, as defined in requirement 81 of Appendix 1C and Sub-appendix 1 (EventFaultType) fault event** with a timestamp equal to the current value of time ~~only if the following two conditions are satisfied: a) the Smart Tachograph is not in calibration mode and b) the vehicle is moving~~.

4.4.4 External GNSS facility certificate expired

GNS_31 If the VU detects that the EGF certificate used for mutual authentication is not valid any longer, the VU ~~shall generate and record a recording equipment fault control device event of type EventFaultType enum ‘56’H ‘1B’H~~ **External GNSS facility certificate expired** with a timestamp equal to the current value of time. The VU shall still use the received GNSS position data.

Figure 64
Schema of the external GNSS facility



5. Vehicle Unit without an external GNSS facility

5.1. Configuration

In this configuration, the GNSS receiver is inside the Vehicle Unit as described in **Figure 144**.

GNS_32 For transmitting position, DOP and satellites data, the GNSS receiver shall act as a talker and transmit NMEA or NMEA-like sentences to the VU processor, which shall act as a listener with a frequency of 1/10 Hz or faster for the pre-defined set of NMEA sentences, which shall include at least the RMC, ~~and GSA~~, AMC and ASA sentences. **Alternatively, the VU processor and the internal GNSS receiver may use other data formats to exchange the data contained in the NMEA or NMEA-like sentences specified in GNS_4, GNS_4a and GNS_5.**

GNS_33 An external GNSS antenna installed on the vehicle or an internal GNSS antenna shall be connected to the VU.

5.2. Transfer of information from the GNSS receiver to the VU

GNS_34 The VU processor checks the received data extracting the information (e.g., latitude, longitude, time) from the RMC NMEA sentence and the AMC sentence.

GNS_35 The RMC NMEA sentence includes the information if the non-authenticated position is valid. If the non-authenticated position is not valid, the position data is not available and it cannot be used to record the position of the vehicle. If the non-authenticated position is valid, the VU processor also extracts the values of HDOP from GSA NMEA.

GNS_36 The VU processor also extracts the information (e.g. latitude, longitude, time) from the AMC sentence. The AMC sentence includes the information if the non-authenticated position is valid according to GNS_4a. If the non-authenticated position is valid, the VU processor also extracts the values of HDOP from ASA NMEA sentences.

5.3. Transfer of information from the VU to the GNSS receiver

GNS_37 The VU processor provides to the GNSS receiver the VU RTC time and the maximal difference between true time and the VU RTC time, in accordance with GNS_3f and GNS_3g.

5.4. Error Handling

5.4.2.1 Absence of position information from GNSS receiver

GNS_384 The VU shall generate and record an absence of position information from GNSS receiver event, as defined in requirement 81 of Appendix 1C and Sub-appendix 1 (EventFaultType) ~~If the VU does not receive data from the GNSS receiver for more than 3 continuous hours, the VU shall generate and record an event of type EventFaultType enum '51'H Internal '0D'H Absence of position information from GNSS receiver event with a timestamp equal to the current value of time only if the following two conditions are satisfied: a) the Smart Tachograph is not in calibration mode and b) the vehicle is moving.~~

6. Position data processing and recording by the VUGNSS Time Conflict

This section is valid both for the configuration of the Smart Tachograph with or without an external GNSS facility. If the VU detects a discrepancy of more than 1 minute between the time of the vehicle unit's time measurement function and the time originating from the GNSS receiver, the VU will record an event of type `EventFaultType` enum '*OB'H Time conflict (GNSS versus VU internal clock)*'. This event is recorded together with the internal clock value of the vehicle unit and comes together with an automatic time adjustment. After a time conflict event has been triggered, the VU will not check the time discrepancy for the next 12 hours. This event shall not be triggered in cases no valid GNSS signal was detectable by the GNSS receiver within the last 30 days. However, when the position information from the GNSS receiver is available again the automatic time adjustment shall be done.

GNS_39 Position data shall be stored in the VU, together with a flag indicating if the position has been authenticated. When position data need to be recorded in the VU, the following rules shall apply:

- a) If both authenticated and standard positions are valid and consistent, the standard position and its accuracy shall be recorded in the VU and the flag shall be set to 'authenticated'.
- b) If both authenticated and standard positions are valid but not consistent, the VU shall store the authenticated position and its accuracy, and the flag shall be set to 'authenticated'.
- c) If the authenticated position is valid and the standard position is not valid, the VU shall record the authenticated position and its accuracy, and the flag shall be set to 'authenticated'.
- d) If the standard position is valid and the authenticated position is not valid, the VU shall record the standard position and its accuracy, and the flag shall be set to 'not authenticated'.

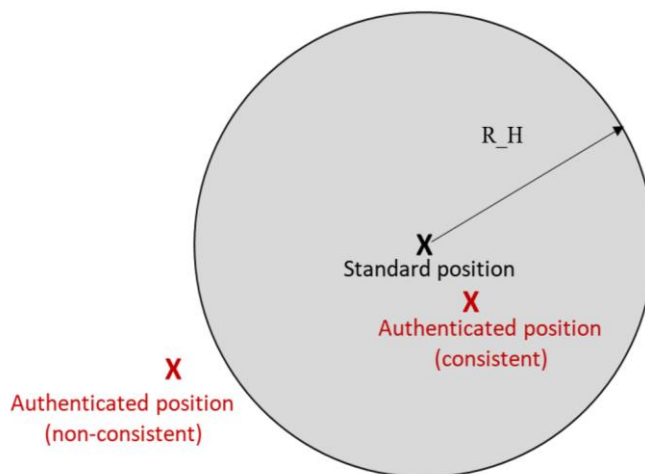
Authenticated and standard positions are considered as consistent, as shown in Figure 7, when the horizontal authenticated position can be found in a circle centered at the horizontal standard position, which radius results of rounding up to the nearest upper whole number the value of R_H calculated according to the following formula:

$$R_H = 1.74 \cdot \sigma_{UERE} \cdot HDOP$$

where:

- R_H is the relative radius of a circle around the estimated horizontal position, in meters. It is an indicator that is used to check consistency between standard and authenticated positions.
- σ_{UERE} is the standard deviation for the user equivalent range error (UERE), which models all measurement errors for the target application, including urban environments. A constant value of $\sigma_{UERE} = 10$ meters shall be used.
- HDOP is the horizontal dilution of precision calculated by the GNSS receiver.
- $\sigma_{UERE} \cdot HDOP$ is the estimation of the root mean squared error in the horizontal domain.

Figure 7
Consistent Authenticated and Standard (non-authenticated) positions



GNS_40 When the value of the Status in a received AMC sentence is set to ‘J’ or ‘O’ or ‘F’ in accordance with requirement GNS_4a, the VU shall generate and record a GNSS anomaly event, as defined in requirement 88a of Appendix 1C and Sub-appendix 1 (EventFaultType). The vehicle unit may perform additional checks before storing a GNSS anomaly event following the reception of a ‘J’ or ‘O’ setting.

7. GNSS Time Conflict

GNS_41 If the VU detects a discrepancy between the time of the vehicle unit’s time measurement function and the time originating from the GNSS signals, it shall generate and record a time conflict event, as defined in requirement 86 of Appendix 1C and Sub-appendix 1 (EventFaultType).

8. Vehicle motion conflict

GNS_4235 The VU shall trigger and record an Vehicle Motion Conflict event (~~see in accordance with requirement 84 of Appendix 1C in this Annex Appendix~~) with a timestamp equal to the current value of time, in case motion information calculated from the motion sensor is contradicted by motion information calculated from the internal GNSS receiver, ~~or from the external GNSS facility,~~ or by other independent motion source(s) as set out in requirement 26 of Appendix 1C.

The vehicle motion conflict event shall be triggered upon occurrence of one of the following trigger conditions:

Trigger condition 1:

~~For the purpose of detecting such contradictions, the~~ the trimmed median value of the speed differences between these sources shall be used, when the position information from the GNSS receiver is available and when the ignition of the vehicle is switched on, as specified below:

- every 10 seconds maximum, the absolute value of the difference between the vehicle speed estimated from the GNSS and the one estimated from the motion sensor shall be computed.

- all the computed values in a time window containing the last five minutes of movement shall be used to compute the **trimmed** median value.
- the **trimmed** median value shall be computed as the average of 80% of the remaining values, after having eliminated the highest ones in absolute values

The Vehicle Motion Conflict event shall be triggered if the **trimmed** median value is above 10 Km/h for five uninterrupted minutes of vehicle movement. ~~Other independent sources of vehicle motion detection may optionally be used, so that a more reliable detection of tachograph manipulations is provided.~~ (Note: the use of the **trimmed** median on the last 5 minutes is applied to mitigate the risk of measurement outliers and transient values).

For the trimmed mean computation, the vehicle shall be considered as moving if at least one vehicle speed value estimated either from motion sensor or from GNSS receiver is not equal to zero. This event shall not be triggered in the following conditions: a) during a ferry/train crossing, b) when the position information from the GNSS receiver shall not be available and c) while in calibration mode.

Trigger condition 2:

The vehicle motion conflict event shall also be triggered if the following condition is true:

$$GnssDistance > [OdometerDifference \times OdometerToleranceFactor + \text{Minimum}(SlipDistanceUpperlimit; (OdometerDifference \times SlipFactor)) + GnssTolerance + FerryTrainDistance]$$

where:

- *GnssDistance* is the distance between the current position of the vehicle and the previous one, both obtained from valid authenticated position messages, without considering the height,
- *OdometerDifference* is the difference between the current odometer value and the odometer value corresponding to the previous valid authenticated position message,
- *OdometerToleranceFactor* is equal to 1.1 (worst case tolerance factor for all measurement tolerances of the vehicle odometer),
- *GnssTolerance* is equal to 1 km (worst case GNSS tolerance),
- *Minimum (SlipDistanceUpperLimit; (OdometerDifference * SlipFactor))* is the minimum value between:
 - *SlipDistanceUpperLimit* which is equal to 10 km (upper limit of the slip distance caused by slipping effects during braking),
 - and *OdometerDifference * SlipFactor*, in which *SlipFactor* is equal to 0.2 (maximal influence of slipping effects during breaking),
 - *FerryTrainDistance* is computed as: $FerryTrainDistance = 200\text{km/h} * t_{FerryTrain}$, where $t_{FerryTrain}$ is the sum of the durations in hours of the ferry/train crossings in the considered time interval. The duration of a ferry/train crossings is defined as the time difference between its end flag and its beginning flag.

The preceding verifications shall be performed every 15 minutes if the necessary position data are available, otherwise as soon as the position data are available.

For this trigger condition:

- **date and time of beginning of event shall be equal to the date and time when the previous position message was received,**
- **date and time of end of event shall be equal to the date and time when the checked condition becomes false again.**

Trigger condition 3:

The vehicle unit encounters a discrepancy consisting of the motion sensor not detecting any movement and the independent motion source detecting movement for a specific period. The conditions to record a discrepancy as well as the period of detection of the discrepancy shall be set out by the vehicle unit manufacturer, although the discrepancy shall be detected in no more than three hours.

Sub-appendix 13. ITS Interface

Table of Content

1.	INTRODUCTION	511
1.1.	SCOPE	511
1.2.	ACRONYMS AND DEFINITIONS	511
2.	SCOPE REFERENCED STANDARDS	511
2.1.	<i>Acronyms, definitions and notations</i>	511
3.	REFERENCED REGULATIONS AND STANDARDS ITS INTERFACE WORKING PRINCIPLES	511
3.1.	COMMUNICATION TECHNOLOGY	512
3.2.	AVAILABLE SERVICES	512
3.3.	ACCESS THROUGH THE ITS INTERFACE	512
3.4.	DATA AVAILABLE AND NEED OF DRIVER CONSENT	514
4.	INTERFACE WORKING PRINCIPLES LIST OF DATA AVAILABLE THROUGH THE ITS INTERFACE AND PERSONAL/NOT PERSONAL CLASSIFICATION	515
4.1.	<i>Preconditions to data transfer via the ITS interface</i>	420
4.1.1	Data provided through the ITS interface	420
4.1.2	Content of the Data	421
4.1.3	ITS Applications	421
4.2.	<i>Communication technology</i>	422
4.3.	<i>PIN authorization</i>	422
4.4.	<i>Message Format</i>	423
4.5.	<i>Driver consent</i>	427
4.6.	<i>Standard data retrieval</i>	428
4.7.	<i>Personal data retrieval</i>	428
4.8.	<i>Event and fault data retrieval</i>	428

1. Introduction

1.1. Scope

ITS_01 This Appendix ~~sub-appendix~~ specifies the **basics of the communication through design and the procedures to follow in order to implement the tachograph interface with Intelligent Transport Systems (ITS) as required in Article 10 of Regulation (EU) no. 16572014 (the regulation).**

ITS_02 The ITS interface shall allow external devices to obtain data from the tachograph, to use tachograph services and also to provide data to the tachograph.

Other tachograph interfaces (e.g. CAN bus) may also be used for that purpose.

This sub-appendix does not specify:

- how data provided through the ITS interface are collected and managed within the tachograph,
- the form of presentation of collected data to applications hosted on the external device,
- the ITS security specification in addition to what provides Bluetooth®,
- the Bluetooth® protocols used by the ITS interface.

1.2. Acronyms and definitions

The following acronyms and definitions specific to this sub-appendix are used:

GNSS Global Navigation Satellite System

ITS Intelligent Transport System

OSI Open Systems Interconnection

VU Vehicle Unit

ITS unit an external device or application using the VU ITS interface.

2. Referenced standards

ITS_03 This sub-appendix refers to and depends upon all or parts of the following regulations and standards. Within the clauses of this sub-appendix, the relevant standards, or relevant clauses of standards, are referred to. In the event of any contradiction the clauses of this sub-appendix shall take precedence.

Standards referenced to in this sub-appendix are:

- Bluetooth® – Core Version 5.0.
- ISO 16844-7: Road vehicles -Tachograph systems - Part 7: Parameters
- ISO/IEC 7498-1:1994 Information technology - Open Systems Interconnection - Basic Reference Model, the Basic Model

3. ITS Interface working principles

ITS_04 The VU shall be responsible to keep updated and maintain tachograph data transmitted through the ITS interface, without any involvement of the ITS interface.

3.1. Communication technology

ITS_05 Communication through the ITS interface shall be performed via Bluetooth® interface and be compatible to Bluetooth® Low Energy according to Bluetooth version 5.0 or newer.

ITS_06 The communication between the VU and the ITS unit shall be established after a Bluetooth® pairing process has been completed.

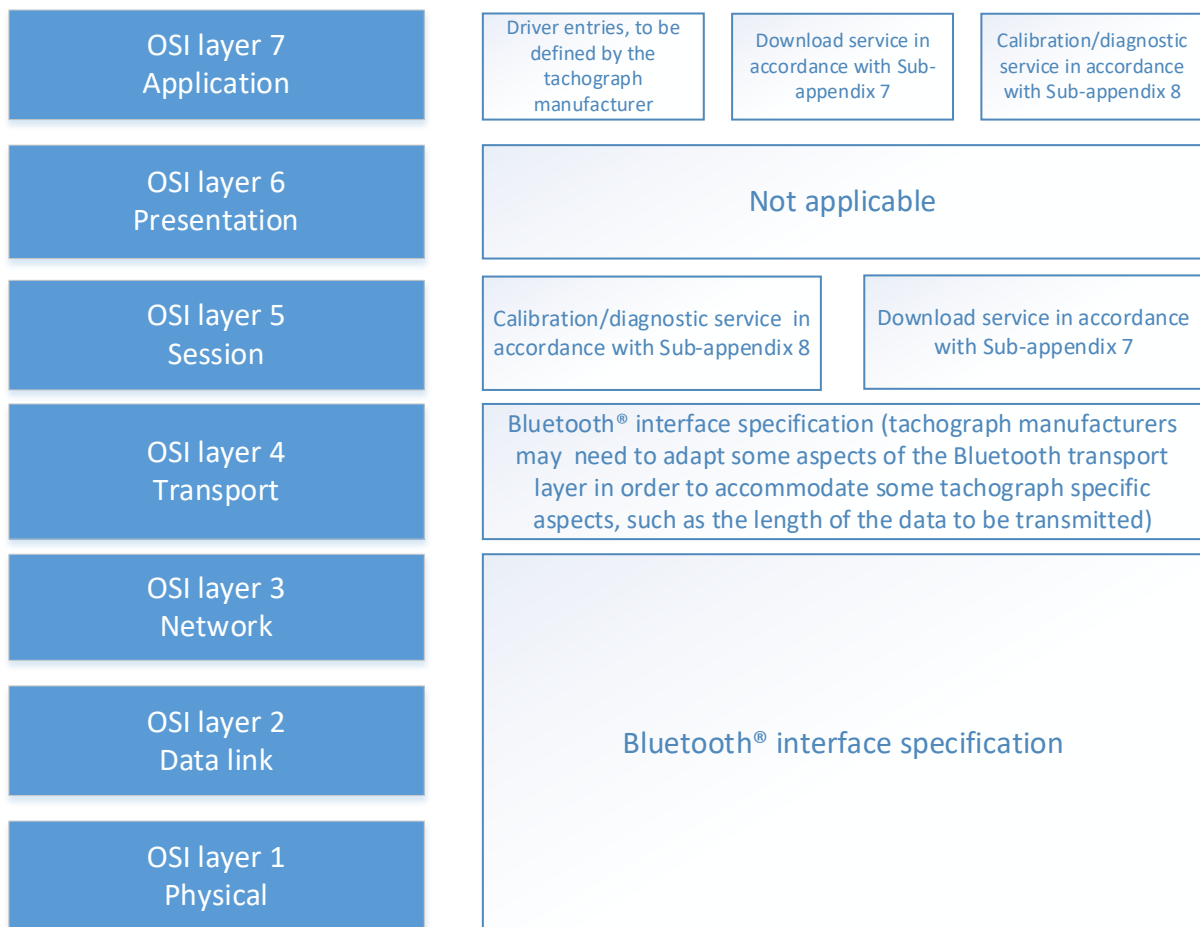
ITS_07 A secure and encrypted communication shall be established between the VU and the ITS unit, in accordance with the Bluetooth® specification mechanisms. This sub-appendix does not specify encryption or other security mechanisms in addition to what Bluetooth® provides.

ITS_08 Bluetooth® is using a server/client model to control the transmission of data between devices, in which the VU shall be the server and the ITS unit shall be the client.

3.2. Available services

ITS_09 The data to be transmitted through the ITS interface in accordance with point 4 shall be made available through the services specified in Sub-appendix 7 and Sub-appendix 8. In addition, the VU shall make available to the ITS unit the services that are necessary for manual data entry in accordance with requirement 61 of Appendix 1C, and optionally, for other data entries in real time.

Partition of the communication through the ITS interface according to the OSI Model layers



ITS_10 When the download interface is used via the front connector, the VU shall not provide the download services specified in Sub-appendix 7 via ITS Bluetooth® connection.

ITS_11 When the calibration interface is used via the front connector, the VU shall not provide the calibration services specified in Sub-appendix 8 via ITS Bluetooth® connection.

3.3. Access through the ITS interface

ITS_12 The ITS interface shall provide a wireless access to all services specified in Sub-appendix 7 and Sub-appendix 8, in replacement of a cable connection to the front connector for calibration and download specified in Sub-appendix 6.

ITS_13 The VU shall make the ITS interface available to the user according to the combination of valid tachograph cards inserted in the VU, as specified in Table 1.

Availability of the ITS interface		Driver slot				
		No card	Driver card	Control card	Workshop card	Company card
Co-driver slot	No card	Not available	Available	Available	Available	Available
	Driver card	Available	Available	Available	Available	Available
	Control card	Available	Available	Available	Not available	Not available
	Workshop card	Available	Available	Not available	Available	Not available

	Company card	Available	Available	Not available	Not available	Available
--	--------------	-----------	-----------	---------------	---------------	-----------

Table 1 - Availability of ITS interface depending on the type of card inserted in the tachograph

ITS_14 After a successful ITS Bluetooth® pairing, the VU shall assign the ITS Bluetooth® connection to the specific inserted tachograph card according to Table 2:

Assignment of the ITS Bluetooth® connection		Driver slot				
		No card	Driver card	Control card	Workshop card	Company card
Co-driver slot	No card	Not available	Driver card	Control card	Workshop card	Company card
	Driver card	Driver card	Driver card (**)	Control card	Workshop card	Company card
	Control card	Control card	Control card	Control card (*)	Not available	Not available
	Workshop card	Workshop card	Workshop card	Not available	Workshop card (*)	Not available
	Company card	Company card	Company card	Not available	Not available	Company card (*)

Table 2 - Assignment of the ITS connection depending on the type of card inserted in the tachograph

(*) The ITS Bluetooth® connection shall be assigned to the tachograph card in the driver slot of the VU.

(**) The user shall select the card to which the ITS Bluetooth® connection shall be assigned (inserted in the driver or in the co-driver slot).

ITS_15 If a tachograph card is withdrawn, then the VU shall terminate the ITS Bluetooth® connection which is assigned to this card.

ITS_16 The VU shall support the ITS connection to at least one ITS unit and may support connections to multiple ITS units at the same time.

ITS_17 The access rights to the data and services available via the ITS interface shall comply with requirements 12 and 13 of Appendix 1C, in addition to the driver consent specified in section 3.4 of this sub-appendix.

3.4. Data available and need of driver consent

ITS_18 All tachograph data available via the services referred to in point 3.3 shall be classified as either personal or not personal for the driver, co-driver or both.

ITS_19 At least the list of data classified as mandatory in section 4 shall be made available through the ITS interface.

ITS_20 The data in section 4 that are classified as ‘personal’ shall only be accessible upon driver consent, accepting therefore that the personal data can leave the vehicle network, except in the case set out in requirement ITS_25, for which the driver consent is not needed.

ITS_21 Data additional to those gathered in point 4 and considered as mandatory may be made available through the ITS interface. Additional data which are not included in point 4 shall be classified as ‘personal’ or not ‘personal’ by the VU manufacturer, being the driver consent requested for those data that have been classified as personal, except in the case set out in requirement ITS_25, for which the driver consent is not needed.

ITS_22 Upon insertion of a driver card which is unknown to the vehicle unit, the cardholder shall be prompted by the tachograph to enter the consent for transmission of personal data output through the ITS interface, in accordance with requirement 61 of Appendix 1C.

ITS_23 The consent status (enabled/disabled) shall be recorded in the data memory of the vehicle unit.

ITS_24 In case of multiple drivers, only the personal data related to the drivers who gave their consent shall be accessible through the ITS interface. For instance, in a crew situation, if only the driver has given his/her consent, personal data related to the co-driver shall not be accessible.

ITS_25 When the VU is in control, company or calibration modes, the access rights through the ITS interface shall be managed according to requirements 12 and 13 of Appendix 1C, hence the driver consent not being needed.

4. List of data available through the ITS interface and personal/not personal classification

Data name	Data format	Source	Data classification (personal/ not personal)		Consent for the availability of the data	Availability
			driver	co-driver		
VehicleIdentificationNumber	Appendix 8	VU	not personal	not personal	no need of consent	mandatory
CalibrationDate	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
TachographVehicleSpeed	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver1WorkingState	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2WorkingState	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
DriveRecognize	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
Driver1TimeRelatedStates	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2TimeRelatedStates	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
DriverCardDriver1	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
DriverCardDriver2	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
OverSpeed	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
TimeDate	Sub-appendix 8	VU	not personal	not personal	no need of consent	mandatory
HighResolutionTotalVehicleDistance	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
HighResolutionTripDistance	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
ServiceComponentIdentification	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
ServiceDelayCalendarTimeBased	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
Driver1Identification	ISO 16844-7	Driver Card	personal	N/A	driver consent	mandatory

Data name	Data format	Source	Data classification (personal/ not personal)		Consent for the availability of the data	Availability
			driver	co-driver		
Driver2Identification	ISO 16844-7	Driver Card	N/A	personal	co-driver consent	mandatory
NextCalibrationDate	Sub-appendix 8	VU	not personal	not personal	no need of consent	mandatory
Driver1ContinuousDrivingTime	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2ContinuousDrivingTime	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
Driver1CumulativeBreakTime	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2CumulativeBreakTime	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
Driver1CurrentDurationOfSelectedActivity	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2CurrentDurationOfSelectedActivity	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
SpeedAuthorised	Sub-appendix 8	VU	not personal	not personal	no need of consent	mandatory
TachographCardSlot1	ISO 16844-7	VU	not personal	N/A	no need of consent	mandatory
TachographCardSlot2	ISO 16844-7	VU	N/A	not personal	no need of consent	mandatory
Driver1Name	ISO 16844-7	Driver Card	personal	N/A	driver consent	mandatory
Driver2Name	ISO 16844-7	Driver Card	N/A	personal	co-driver consent	mandatory
OutOfScopeCondition	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
ModeOfOperation	ISO 16844-7	VU	not personal	not personal	no need of consent	mandatory
Driver1CumulatedDrivingTimePreviousAndCurrentWeek	ISO 16844-7	VU	personal	N/A	driver consent	mandatory
Driver2CumulatedDrivingTimePreviousAndCurrentWeek	ISO 16844-7	VU	N/A	personal	co-driver consent	mandatory
EngineSpeed	ISO 16844-7	VU	personal	N/A	driver consent	optional
RegisteringMemberState	Sub-appendix 8	VU	not personal	not personal	no need of consent	mandatory
VehicleRegistrationNumber	Sub-appendix 8	VU	not personal	not personal	no need of consent	mandatory
Driver1EndOfLastDailyRestPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2EndOfLastDailyRestPeriod	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1EndOfLastWeeklyRestPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2EndOfLastWeeklyRestPeriod	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1EndOfSecondLastWeeklyRestPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2EndOfSecondLastWeeklyRestPeriod	ISO 16844-7	VU	N/A	Personal	co-driver consent	optional
Driver1TimeLastLoadUnloadOperation	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2TimeLastLoadUnloadOperation	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1CurrentDailyDrivingTime	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2CurrentDailyDrivingTime	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1CurrentWeeklyDrivingTime	ISO 16844-7	VU	personal	N/A	driver consent	optional

Data name	Data format	Source	Data classification (personal/ not personal)		Consent for the availability of the data	Availability
			driver	co-driver		
Driver2CurrentWeeklyDrivingTime	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1TimeLeftUntilNewDailyRestPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2TimeLeftUntilNewDailyRestPeriod	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1CardExpiryDate	ISO 16844-7	Driver Card	personal	N/A	driver consent	optional
Driver2CardExpiryDate	ISO 16844-7	Driver Card	N/A	personal	co-driver consent	optional
Driver1CardNextMandatoryDownloadDate	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2CardNextMandatoryDownloadDate	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
TachographNextMandatoryDownloadDate	ISO 16844-7	VU	not personal	not personal	no need of consent	optional
Driver1TimeLeftUntilNewWeeklyRestPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2TimeLeftUntilNewWeeklyRestPeriod	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1NumberOfTimes9hDailyDrivingTimes Exceeded	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2NumberOfTimes9hDailyDrivingTimes Exceeded	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1CumulativeUninterruptedRestTime	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2CumulativeUninterruptedRestTime	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1MinimumDailyRest	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2MinimumDailyRest	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1MinimumWeeklyRest	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2MinimumWeeklyRest	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1MaximumDailyPeriod	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2MaximumDailyPeriod	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1MaximumDailyDrivingTime	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2MaximumDailyDrivingTime	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1NumberOfUsedReducedDailyRestPeriods	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2NumberOfUsedReducedDailyRestPeriods	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
Driver1RemainingCurrentDrivingTime	ISO 16844-7	VU	personal	N/A	driver consent	optional
Driver2RemainingCurrentDrivingTime	ISO 16844-7	VU	N/A	personal	co-driver consent	optional
VehiclePosition	Sub-appendix 8	VU	personal	personal	driver and co-driver consent	mandatory
ByDefaultLoadType	Sub-appendix 8	VU	personal	personal	driver and co-driver consent	mandatory

The Regulation specify that the tachographs of vehicles may be equipped with standardised interfaces allowing the data recorded or produced by tachograph to be used in operational mode, by an external device, provided that the following conditions are met:

- (a) — the interface does not affect the authenticity and the integrity of the data of the tachograph;

(b) ~~the interface complies with the detailed provisions of Article 11 of the Regulation this sub-appendix;~~

(c) ~~the external device connected to the interface has access to personal data, including ge positioning data, only after the verifiable consent of the driver to whom the data relates.~~

~~2. Scope~~

The scope of this Appendix ~~sub-appendix~~ is to specify how applications hosted on external devices can via a Bluetooth® connection obtain data (*the Data*) from a tachograph.

~~The Data available via this interface is described in the Annex 1 of the present document. This interface does not prohibit the implementation of other interfaces (e.g. via the CAN bus) to transmit the data of the VU to other vehicle processing units.~~

~~— This Appendix **sub-appendix** specifies:~~

~~— *The Data* available through the ITS interface~~

~~— The Bluetooth® profile that is used to transfer the data~~

~~— The enquiry and download procedures and sequence of operations~~

~~— The pairing mechanism between the tachograph and the external device~~

~~— The consent mechanism available to the driver~~

~~— For clarification, this Appendix **Sub-appendix** does not specify:~~

~~— The collection of *the Data* operation and management within the VU (which shall be specified elsewhere within the Regulation **this Agreement** or otherwise shall be a function of product design).~~

~~— The form of presentation of collected data to application hosted on the external device.~~

~~— Data security provisions above what provides Bluetooth® (such as encryption) concerning the content of *the Data* (which shall be specified elsewhere within *the Regulation* [Appendix 10 **Sub-appendix** 11 Common Security Mechanisms]).~~

~~— The Bluetooth® protocols used by the ITS interface~~

~~2.1. Acronyms, definitions and notations~~

The following acronyms and definitions specific to this Appendix ~~sub-appendix~~ are used in this ~~sub~~ appendix:

~~*the Communication* exchange of information/data between a master unit (i.e. the tachographs) and an external unit through the ITS interface over Bluetooth®.~~

~~*the Data* — Data sets as specified in Annex 1.~~

~~the Regulation — Regulation (EU) no 165/2014 of the European Parliament and the Council of 4 February 2014 on the tachograph in road transport, repealing Council Regulation (EEC) No 561/2006 of the European Parliament and the Council on the harmonisation of certain social legislation relating to road transport.~~

~~BR — Basic Rate~~

~~EDR — Enhanced Data Rate~~

GNSS	Global Navigation Satellite System
IRK	Identity Resolution Key
ITS	Intelligent Transport System
LE	Low Energy
PIN	Personal Identification Number
PUC	Personal Unblocking Code
SID	Service Identifier
SPP	Serial Port Profile
SSP	Secure Simple Pairing
TRTP	Transfer Request Parameter
TREP	Transfer Response Parameter
VU	Vehicle Unit

3. Referenced Regulation and Standards

The specification defined in this Appendix ~~sub-appendix~~ refers to and depends upon all or parts of the following regulations and standards. Within the clauses of this Appendix ~~sub-appendix~~ the relevant standards, or relevant clauses of standards, are specified. In the event of any contradiction the clauses of this Appendix ~~sub-appendix~~ shall take precedence.

Regulations and **Standards** referenced in this Appendix ~~sub-appendix~~ are:

Regulation (EU) No 165/2014 of the European Parliament and the Council of 4 February 2014 on tachographs in road transport repealing Council Regulation (EEC) No. 3821/85 on recording equipment in road transport and amending Regulation (EC) no 561/2006 of the European Parliament and of the Council on the harmonisation of relating to road transport.

Regulation (EU) no. 561/2006 of the European Parliament and the Council of 15 March 2006 on the harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) no. 3821/85 and (EC) No. 2135/98 and repealing Council Regulation (EEC) No. 3820/85.

- ISO 16844 – 4 : Road vehicles – Tachograph systems – Part 4: Can interface
- ISO 16844 – 7 : Road vehicles – Tachograph systems – Part 7: Parameters
- Bluetooth® – Serial Port Profile – V1.2
- Bluetooth® – Core Version 4.2
- NMEA 0183 V4.1 protocol

4. Interface working principles

4.1. Preconditions to data transfer via the ITS interface

The VU shall be responsible to keep updated and maintain the data to be stored in the VU, without any involvement of the ITS interface. The means by which this is achieved is internal to the VU, specified elsewhere in the Regulation ~~this Agreement~~, and is not specified in this Appendix ~~sub-appendix~~.

~~4.1.1 Data provided through the ITS interface~~

~~The VU shall be responsible to update the data that will be available through the ITS interface at a frequency determined within VU procedures, without any involvement of ITS interface. The VU data shall be used as a basis to populate and update *the Data*, the means by which this is achieved is specified elsewhere in the Regulation **this Agreement** or if there is no such specification is a function of product design and is not specified in this Appendix **sub-appendix**.~~

~~4.1.2 Content of the Data~~

~~The content of *the Data* shall be as specified in Annex 1 of this **sub** appendix.~~

~~4.1.3 ITS Applications~~

~~ITS applications will be using the data made available through the ITS interface for instance to optimize driver activities management while respecting the Regulations **provisions of this Agreement**, to detect possible faults of the tachograph or to use the GNSS data. The specification of the applications is not within the scope of this Appendix **sub-appendix**.~~

~~**Contracting Parties may set up restrictions to the transmission of data by ITS applications; those restrictions shall not affect the data provided through the ITS interface in accordance with point 4.1.1. Contracting Parties shall abide by the legislation on data protection in force in their respective territories, in what respects to collection, storage, processing and use of personal data using ITS.**~~

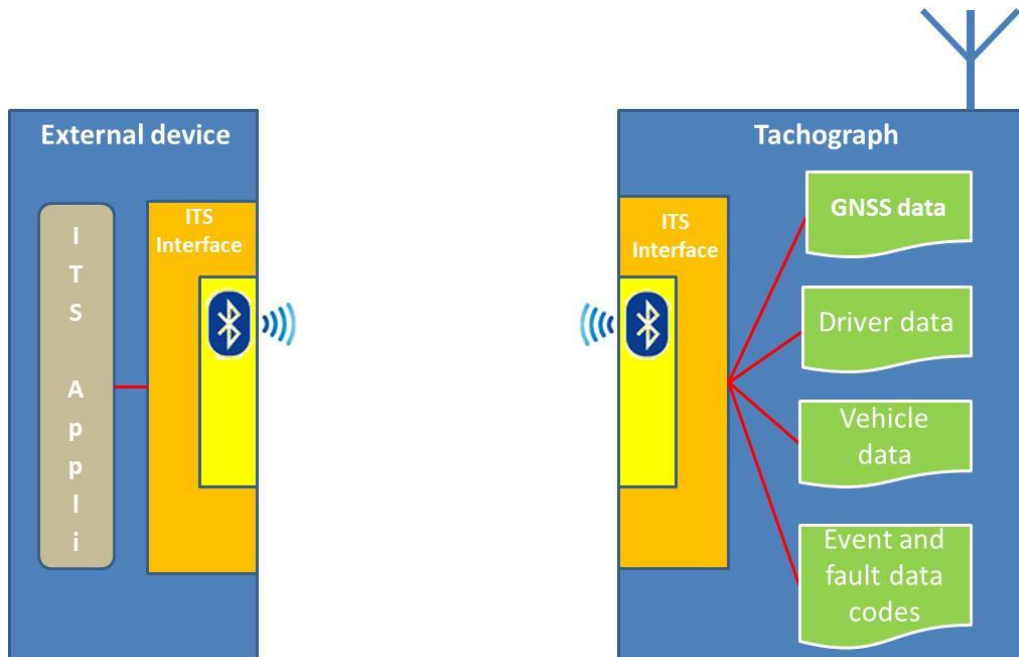
~~4.2. Communication technology~~

~~*The Data* exchange using the ITS interface shall be performed via a Bluetooth® interface compatible via version 4.2 or later. Bluetooth® operates in the unlicensed industrial, scientific and medical (ISM) band at 2.4 to 2.485 GHz. Bluetooth® 4.2 offers enhanced privacy and security mechanisms and increases speed and reliability of data transfers. For the purpose of this specification is Bluetooth® class 2 radio used with a range up to 10 meters. More information on Bluetooth® 4.2 is available on [www.bluetooth.com](https://www.bluetooth.com/https://www.bluetooth.org/en-us/specification/adopted-specifications?_ga=1.215147412.2083380574.1435305676) (https://www.bluetooth.org/en-us/specification/adopted-specifications?_ga=1.215147412.2083380574.1435305676).~~

~~*The Communication* shall be established with the communications equipment after a pairing process has been completed by an authorized device. As Bluetooth® is using a master/slave model to control when and where devices can send data, the tachograph will play the role of master while the external device will be the slave.~~

~~When an external device comes within range of the VU for the first time, the Bluetooth® pairing process can be initiated (see also annex 2). The devices share their addresses, names, and profiles and common secret key, which allows them to bond whenever they're **they are** together in the future. Once this step is completed, the external device is trusted and is in state to initiate requests to download data from the tachograph. It is not foreseen to add encryption mechanisms beyond what Bluetooth® provides. However, if additional security mechanisms are needed, this will be done in accordance with Appendix 10 **Sub-appendix 11** Common Security Mechanisms.~~

The overall communication principle is described in the following figure.



The SPP (Serial Port Profile) profile of Bluetooth® shall be used to transfer data from the VU to the external device.

4.3. PIN authorization

For security reasons, the VU shall enforce ~~will require~~ a PIN code authorization system separated from the Bluetooth pairing. Each VU shall be able to generate PIN codes for authentication purposes composed of at least 4 digits. Every time an external device pairs with the VU, it must provide the correct PIN code before receiving any data.

Succeeding entering the PIN shall result in putting the device on the whitelist. The whitelist shall store at least 64 devices paired with the particular VU.

Failing to provide the correct PIN code three times in a row shall result in putting temporarily the device on the blacklist. While blacklisted, every new attempt from the device shall be rejected. Further ~~§§§§§~~ to provide the correct PIN code three times in a row shall result in increasingly longer ban duration (See table 1). Providing the correct PIN code shall reset the ban duration and the number of attempt. Figure 1 in Annex 2 represents the sequence diagram of a PIN validation attempt.

Table 150

Ban duration depending on the number of consecutive failure to provide the correct PIN code

<i>Number of consecutive failure</i>	<i>Ban duration</i>
3	30 seconds
6	5 minutes
9	1 hour
12	24 hours

<i>Number of consecutive failure</i>	<i>Ban duration</i>
15	Permanent

Failing to provide the correct PIN code fifteen times (5x3) in a row shall result in a permanent blacklisting of the ITS Unit. Only providing the correct PUC code shall overturn this permanent ban.

The PUC code shall be composed of 8 digits and provided by the manufacturer with the VU. Failing to provide the correct PUC code ten times in a row will irrevocably blacklist the ITS Unit.

While the manufacturer may offer an option to change the PIN code directly through the VU, the PUC code shall not be alterable. Modifying the PIN code, if possible, shall require to enter the current PIN code directly in the VU.

Furthermore any devices stored in the whitelist shall be kept until manual removal of by the user (e.g. via the man-machine interface of the VU or other means). By doing so lost or stolen ITS units may be removed from the whitelist. Also, any ITS Unit leaving the Bluetooth connection range for more than 24 hours shall be automatically removed from the VU whitelist and must provide the correct PIN code again when the connection is established again.

The format of the messages between the VU interface and the VU are not provided but left to the discretion of the manufacturer. Said manufacturer shall however ensure the message format between the ITS Unit and the VU interface is respected (see ASN.1 specifications).

Any data request shall thus be met with the proper verification of the sender's credential before any form of treatment. Figure 2 of Annex 2 represents the sequence diagram for this procedure. Any blacklisted device shall receive an automatic rejection, any non blacklisted non-whitelisted device shall receive a PIN request it needs to fulfill before resending its data request.

4.4. Message Format

All messages exchanged between the ITS Unit and the VU interface shall be formatted with a structure consisting of three parts: A header composed by a target byte (TGT), a source byte (SRC) and a length byte (LEN).

The data field composed by a service identifier byte (SID) and a variable amount of data bytes (maximum 255).

The checksum byte is the 1 byte sum series modulo 256 of all the bytes of the message excluding the CS itself.

The message shall be Big Endian.

Table 2-51
General message format

<i>Header</i>			<i>Data Field</i>				<i>Checksum</i>	
TGT	SRC	LEN	SID	TRTP	CC	CM	DATA	CS
3 bytes			Max. 255 bytes				1 byte	

Header

TGT and SRC : the ID of the Target (TGT) and Source (SRC) devices of the message. The VU Interface shall have the default ID "EE". This ID cannot be changed. The ITS Unit shall

use the default ID “A0” for its first message of the communication session. The VU Interface shall then assign an unique ID to the ITS Unit and informs it of this ID for future messages during the session.

The LEN byte shall only take into account the “DATA” part of the Data Field (see Table 2), the 4 first bytes are implicit.

The VU Interface shall confirm the authenticity of the message’s sender by cross checking its own IDList with the Bluetooth data by checking the ITS Unit listed at the provided ID is currently in the range of the Bluetooth connection.

Data Field

Besides the SID, the Data Field shall also contain other parameters: a transfer request parameter (TRTP) and Counter bytes.

If the data which need to be carried ~~handled~~ is too long ~~larger~~ than the available space in one message, it will be split in several submessages. Each submessage shall have the same Header and SID, but will contain a 2 bytes counter, Counter Current (CC) and Counter Max (CM), to indicate the submessage number. To enable error checking and abort the receiving device acknowledges every submessage. The receiving device can accept the submessage, ask for it to be re transmitted, request the sending device to start again or abort the transmission.

If not used, CC and CM shall be given the value 0xFF.

For instance, the following message

HEADER	SID	TRTP	CC	CM	DATA	CS
3 bytes			Longer than 255 bytes			1 byte

Shall be transmitted as such:

HEADER	SID	TRTP	01	n	DATA	CS
3 bytes			255 bytes			1 byte

HEADER	SID	TRTP	02	n	DATA	CS
3 bytes			255 bytes			1 byte

HEADER	SID	TRTP	N	N	DATA	CS
3 bytes			Max. 255 bytes			1 byte

Table 3 contains the messages the VU and the ITS Unit shall be able to exchange. The content of each parameter is given in hexadecimal. Aren’t represented in the table CC and CM for clarity, see above for complete format

Table 3 52

Detailed message content

Message	Header		LEN	DATA			Checksum
	TGT	SRC		SID		DATA	
<i>RequestPIN</i>	<i>ITSID</i>	EE	00	01	FF		
<i>SendITSID</i>	<i>ITSID</i>	EE	01	02	FF	<i>ITSID</i>	
<i>SendPIN</i>	EE	<i>ITSID</i>	04	03	FF	4*INTEGER (0..9)	
<i>PairingResult</i>	<i>ITSID</i>	EE	01	04	FF	BOOLEAN (T/F)	

Message	Header		DATA			Checksum
	TGT	SRC	LEN	SID	DATA	
<i>SendPUC</i>	EE	<i>ITSID</i>	08	05	FF	8*INTEGER(0..9)
<i>BanLiftingResult</i>	<i>ITSID</i>	EE	04	06	FF	BOOLEAN (T/F)
<i>RequestRejected</i>	<i>ITSID</i>	EE	08	07	FF	Time
<i>RequestData</i>						
standardTachData	EE	<i>ITSID</i>	04	08	01	
personalTachData	EE	<i>ITSID</i>	04	08	02	
gnssData	EE	<i>ITSID</i>	04	08	03	
standardEventData	EE	<i>ITSID</i>	04	08	04	
personalEventData	EE	<i>ITSID</i>	04	08	05	
standardFaultData	EE	<i>ITSID</i>	04	08	06	
manufacturerData	EE	<i>ITSID</i>	04	08	07	
<i>RequestAccepted</i>	<i>ITSID</i>	EE	Len	09	TREP	Data
<i>DataUnavailable</i>						
No data available	<i>ITSID</i>	EE	02	0A	TREP	10
Personal data not shared	<i>ITSID</i>	EE	02	0A	TREP	11
<i>NegativeAnswer</i>						
General reject	<i>ITSID</i>	EE	02	0B	SID-Req10	
Service not supported	<i>ITSID</i>	EE	02	0B	SID-Req11	
Sub-function not supported	<i>ITSID</i>	EE	02	0B	SID-Req12	
Incorrect message length	<i>ITSID</i>	EE	02	0B	SID-Req13	
Conditions not correct or request sequence error	<i>ITSID</i>	EE	02	0B	SID-Req22	
Request out of range	<i>ITSID</i>	EE	02	0B	SID-Req31	
Response pending	<i>ITSID</i>	EE	02	0B	SID-Req78	
ITSID Mismatch	<i>ITSID</i>	EE	02	0B	SID-ReqFC	
ITSID Not Found	<i>ITSID</i>	EE	02	0B	SID-ReqFB	

RequestPIN (SID-01)

This message is issued by the VU Interface if a non-blacklisted but non-whitelisted ITS unit is sending any data request.

SendITSID (SID-02)

This message is issued by the VU Interface whenever a new device is sending a request. This device shall use the default ID "A0" before getting assigned an unique ID for the communication session.

SendPIN (SID-03)

This message is issued by the ITS Unit to be whitelisted from the VU interface. The content of this message is a 4 INTEGER between 0 and 9 code.

PairingResult (SID-04)

This message is issued by the VU Interface to inform the ITS Unit if the PIN code it sent was correct. The content of this message shall be a BOOLEAN with the value “True” if the PIN code was correct and “False” otherwise.

SendPUC (SID-05)

This message is issued by the ITS Unit to lift a blacklist sanction from the VU interface. The content of this message is a 8 INTEGER between 0 and 9 code.

BanLiftingResult (SID-06)

This message is issued by the VU Interface to inform the ITS Unit if the PUC code it sent was correct. The content of this message shall be a BOOLEAN with the value “True” if the PUC code was correct and “False” otherwise.

RequestRejected (SID-07)

This message is issued by the VU Interface as a reply to any message from a blacklisted ITS Unit except “SendPUC”. The message shall contain the remaining time the ITS Unit is blacklisted, following the “Time” sequence format as defined in Annex 3.

RequestData (SID-08)

This message for data accessing is issued by the ITS Unit. A one byte transfer request parameter (TRTP) indicates the type of data required. There are several types of data:

- ~~standardTachData (TRTP 01): Data available from the tachograph classified as non-personal.~~
- ~~personalTachData (TRTP 02): Data available from the tachograph classified as personal.~~
- ~~gnssData (TRTP 03): GNSS data, always personal.~~
- ~~standardEventData (TRTP 04): Recorded event data classified as non-personal.~~
- ~~personalEventData (TRTP 05): Recorded event data classified as personal.~~
- ~~standardFaultData (TRTP 06): Recorded faults classified as non-personal.~~
- ~~manufacturerData (TRTP 07): data made available by the manufacturer.~~

See Annex 3 of this ~~sub~~ appendix for more information about the content of each data type.

See Appendix ~~Sub~~ **appendix 12** for more information about the format and content of GNSS data.

See Annex 1B ~~Appendix 1B~~ and **1C** for more information about event data code and faults.

ResquestAccepted (SID-09)

This message is issued by the VU Interface if a ITS Unit “RequestData” message has been accepted. This message contains a 1 byte TREP, which is the TRTP byte of the associated RequestData message, and all the data of the requested type.

DataUnavailable (SID 0A)

This message is issued by the VU Interface if, for a certain reason, the requested data aren't available to be sent to a whitelisted ITS Unit. The message contains a 1byte TRTP which is the TRTP of the required data and a 1 byte error code specified in the table 3. The Following codes are available:

- ~~No data available (10): The VU interface can't access the VU data for unspecified reasons.~~
- ~~Personal data not shared (11): The ITS Unit tries to retrieve personal data when they are not shared.~~

NegativeAnswer (SID 0B)

These messages are issued by the VU Interface if a request cannot be completed for any other reason than the unavailability of the data. These messages are typically the result of a bad request format (Length, SID, ITSID...) but aren't limited to that. The TRTP in the Data Field contains the SID of the request. The Data Field contains a code identifying the reason of the negative answer. The following codes are available:

- ~~General Reject (code: 10)~~
- ~~The action can't be performed for a reason which isn't cited below nor in section (Enter *DataUnavailable* section number).~~
- ~~Service not supported (code: 11)~~
- ~~The request's SID isn't understood.~~
- ~~Sub function not supported (code: 12)~~
- ~~The request's TRTP isn't understood. It can be for instance missing or out of accepted values.~~
- ~~Incorrect message length (code: 13)~~
- ~~The length of the received message is wrong (mismatch between the LEN byte and the actual message length).~~
- ~~Conditions not correct or request sequence error (code: 22)~~
- ~~The required service is not active or the sequence of request messages is not correct~~
- ~~Request out of range (code: 33)~~
- ~~The request parameter record (data field) is not valid~~
- ~~Response pending (code: 78)~~
- ~~The action requested cannot be completed in time and the VU is not ready to accept another request.~~
- ~~ITSID Mismatch (code: FB)~~
- ~~The SRC ITSID doesn't match the associated device after comparison with the Bluetooth information.~~
- ~~ITSID Not Found (code: FC)~~

The SRC ITSID isn't associated with any device.

Lines 1 through 72 (**FormatMessageModule**) of the ASN.1 code in Annex 3 specify the messages format as described in table 3. More details about the messages content is given below.

4.5. Driver consent

All the data available are classified as either standard or personal. Personal data shall only be accessible if the driver gave his/her consent, accepting his/her tachograph personal data can leave the vehicle network for third party applications.

Driver consent is given when, at first insertion of a given driver card or workshop card currently unknown to the vehicle unit, the cardholder is invited to express his consent for tachograph related personal data output through the optional ITS interface. (see also Annex 1C Appendix 1C paragraph 3.6.2).

The consent status (enabled/disabled) is recorded in the memory of the tachograph.

In case of multiple drivers, only the personal data about the drivers who gave their consent shall be shared with the ITS interface. For instance, if there's two drivers in the vehicle, and only the first driver accepted to share his personal data, the ones concerning the second driver shall not be shared.

4.6. Standard data retrieval

Figure 3 of Annex 2 represents the sequence diagrams of a valid request sent by the ITS Unit to access standard data. The ITS Unit is properly whitelisted and isn't requesting personal data, no further verification is required. The diagrams consider the proper procedure illustrated in Figure 2 of Annex 2 has already been followed. They can be equated to the *REQUEST TREATMENT* gray box of Figure 2.

Amongst available data, shall be considered standard:

- standardTachData (TRTP 01)
- StandardEventData (TRTP 04)
- standardFaultData (TRTP 06)

4.7. Personal data retrieval

Figure 4 of Annex 2 represents the sequence diagram for personal data request processing. As previously stated, the VU interface shall only send personal data if the driver has given his explicit consent (see also 4.5). Otherwise, the request must be automatically rejected.

Amongst available data, shall be considered personal:

- personalTachData (TRTP 02)
- gnsData (TRTP 03)
- personalEventData (TRTP 05)
- manufacturerData (TRTP 07)

4.8. Event and fault data retrieval

ITS units shall be able to request events data containing the list of all the unexpected events. These data are considered standard or personal, see Annex 3. The content of each event is in accordance with the documentation provided in Annex 1 of this sub appendix.

ANNEX 1

~~(1) LIST OF AVAILABLE DATA THROUGH THE ITS INTERFACE~~

~~(2) CONTINUOUS GNSS DATA AVAILABLE AFTER DRIVER
CONSENT~~

~~See Sub-appendix 12—GNSS.~~

(3) EVENT CODES AVAILABLE WITHOUT DRIVER CONSENT

Data	Source	Data classification (personal/not personal)
VehicleIdentificationNumber	Vehicle Unit	not personal
CalibrationDate	Vehicle Unit	not personal
TachographVehicleSpeed speed instant t	Vehicle Unit	personal
Driver1WorkingState Selector driver	Vehicle Unit	personal
Driver2WorkingState	Vehicle Unit	personal
DriveRecognize Speed Threshold detected	Vehicle Unit	not personal
Driver1TimeRelatedStates Weekly day time	Driver Card	personal
Driver2TimeRelatedStates	Driver Card	personal
DriverCardDriver1	Vehicle Unit	not personal
DriverCardDriver2	Vehicle unit	not personal
OverSpeed	Vehicle Unit	personal
TimeDate	Vehicle Unit	not personal
HighResolutionTotalVehicleDistance	Vehicle Unit	not personal
ServiceComponentIdentification	Vehicle Unit	not personal
ServiceDelayCalendarTimeBased	Vehicle Unit	not personal
Driver1Identification	Driver Card	personal
Driver2Identification	Driver Card	personal
NextCalibrationDate	Vehicle Unit	not personal
Driver1ContinuousDrivingTime	Driver Card	personal
Driver2ContinuousDrivingTime	Driver Card	personal
Driver1CumulativeBreakTime	Driver Card	personal
Driver2CumulativeBreakTime	Driver Card	personal
Driver1CurrentDurationOfSelectedActivity	Driver Card	personal
Driver2CurrentDurationOfSelectedActivity	Driver Card	personal
SpeedAuthorised	Vehicle Unit	not personal
TachographCardSlot1	Driver Card	not personal
TachographCardSlot2	Driver Card	not personal
Driver1Name	Driver Card	personal
Driver2Name	Driver Card	personal
OutOfScopeCondition	Vehicle Unit	not personal
ModeOfOperation	Vehicle Unit	not personal
Driver1CumulatedDrivingTimePreviousAndCurrentWeek	Driver Card	personal
Driver2CumulatedDrivingTimePreviousAndCurrentWeek	Driver Card	personal
EngineSpeed	Vehicle Unit	personal
RegisteringMemberState	Vehicle Unit	not personal
VehicleRegistrationNumber	Vehicle Unit	not personal
Driver1EndOfLastDailyRestPeriod	Driver Card	personal
Driver2EndOfLastDailyRestPeriod	Driver Card	personal
Driver1EndOfLastWeeklyRestPeriod	Driver Card	personal
Driver2EndOfLastWeeklyRestPeriod	Driver Card	personal
Driver1EndOfSecondLastWeeklyRestPeriod	Driver Card	personal
Driver2EndOfSecondLastWeeklyRestPeriod	Driver Card	personal
Driver1CurrentDailyDrivingTime	Driver Card	personal
Driver2CurrentDailyDrivingTime	Driver Card	personal
Driver1CurrentWeeklyDrivingTime	Driver Card	personal
Driver2CurrentWeeklyDrivingTime	Driver Card	personal
Driver1TimeLeftUntilNewDailyRestPeriod	Driver Card	personal
Driver2TimeLeftUntilNewDailyRestPeriod	Driver Card	personal
Driver1CardExpiryDate	Driver Card	personal
Driver2CardExpiryDate	Driver Card	personal
Driver1CardNextMandatoryDownloadDate	Driver Card	personal
Driver2CardNextMandatoryDownloadDate	Driver Card	personal
TachographNextMandatoryDownloadDate	Vehicle Unit	not personal
Driver1TimeLeftUntilNewWeeklyRestPeriod	Driver Card	personal
Driver2TimeLeftUntilNewWeeklyRestPeriod	Driver Card	personal
Driver1NumberOfTimes9hDailyDrivingTimesExceeded	Driver Card	personal
Driver2NumberOfTimes9hDailyDrivingTimesExceeded	Driver Card	personal
Driver1CumulativeUninterruptedRestTime	Driver Card	personal
Driver2CumulativeUninterruptedRestTime	Driver Card	personal
Driver1MinimumDailyRest	Driver Card	personal
Driver2MinimumDailyRest	Driver Card	personal
Driver1MinimumWeeklyRest	Driver Card	personal
Driver2MinimumWeeklyRest	Driver Card	personal
Driver1MaximumDailyPeriod	Driver Card	personal
Driver2MaximumDailyPeriod	Driver Card	personal
Driver1MaximumDailyDrivingTime	Driver Card	personal
Driver2MaximumDailyDrivingTime	Driver Card	personal
Driver1NumberOfUsedReducedDailyRestPeriods	Driver Card	personal
Driver2NumberOfUsedReducedDailyRestPeriods	Driver Card	personal
Driver1RemainingCurrentDrivingTime	Driver Card	personal
Driver2RemainingCurrentDrivingTime	Driver Card	personal
GnssPosition	Vehicle Unit	personal

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Insertion of a non-valid card	— the 10 most recent events.	— date and time of event, — card(s) type, number, issuing Member State Contracting Party and generation of the card creating the event. — number of similar events that day
Card conflict	— the 10 most recent events.	— date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of the two cards creating the conflict.
Last card session not correctly closed	— the 10 most recent events.	— date and time of card insertion, — card(s) type, number, issuing Member State Contracting Party and generation, — last session data as read from the card: — date and time of card insertion, — VRN, Contracting Party of registration and VU generation.
Power supply interruption (2)	— the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days.	— date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Communication error with the remote communication facility	— the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days.	— date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Absence of position information from GNSS receiver	— the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days.	— date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Communication error with the external GNSS facility	— the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days.	— date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Motion data error	<ul style="list-style-type: none"> — the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> — date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Vehicle motion conflict	<ul style="list-style-type: none"> — the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> — date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Security breach attempt	<ul style="list-style-type: none"> — the 10 most recent events per type of event. 	<ul style="list-style-type: none"> — date and time of beginning of event, — date and time of end of event (if relevant), — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — type of event.
Time conflict	<ul style="list-style-type: none"> — the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> — recording equipment control device date and time — GNSS date and time, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.

(4) EVENT CODES AVAILABLE WITH DRIVER CONSENT

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Driving without an appropriate card	<ul style="list-style-type: none"> — the longest event for each of the 10 last days of occurrence, — the 5 longest events over the last 365 days. 	<ul style="list-style-type: none"> — date and time of beginning of event, — date and time of end of event, — card(s) type, number, issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the event, — number of similar events that day.
Card insertion while driving	<ul style="list-style-type: none"> — the last event for each of the 10 last days of occurrence, 	<ul style="list-style-type: none"> — date and time of the event, — card(s) type, number, issuing Member State Contracting Party and generation, — number of similar events that day

<i>Event</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
Over speeding (1)	<ul style="list-style-type: none"> _____ the most serious event for each of the 10 last days of occurrence (i.e. the one with the highest average speed), _____ the 5 most serious events over the last 365 days. _____ the first event having occurred after the last calibration 	<ul style="list-style-type: none"> _____ date and time of beginning of event, _____ date and time of end of event, _____ maximum speed measured during the event, _____ arithmetic average speed measured during the event, _____ card type, number, issuing Member State Contracting Party and generation of the driver card (if applicable), _____ number of similar events that day.

(5) FAULT DATA CODES AVAILABLE WITHOUT DRIVER CONSENT

<i>Fault</i>	<i>Storage rules</i>	<i>Data to be recorded per fault</i>
Card fault	<ul style="list-style-type: none"> _____ the 10 most recent driver card faults. 	<ul style="list-style-type: none"> _____ date and time of beginning of fault, _____ date and time of end of fault, _____ card(s) type, number, issuing Member State Contracting Party and generation.
Control device faults	<ul style="list-style-type: none"> _____ the 10 most recent faults for each type of fault, _____ the first fault after the last calibration. 	<ul style="list-style-type: none"> _____ date and time of beginning of fault, _____ date and time of end of fault, _____ type of fault, _____ card(s) type, number and issuing Member State Contracting Party and generation of any card inserted at beginning and/or end of the fault.

This fault shall be triggered for any of these failures, while not in calibration mode:

- VU internal fault
- Printer fault
- Display fault
- _____ Downloading fault
- _____ Sensor fault
- _____ GNSS receiver or external GNSS facility fault
- _____ Remote Communication facility fault
- _____ ITS interface fault (if applicable)

(6) MANUFACTURER SPECIFIC EVENTS AND FAULTS WITHOUT DRIVER CONSENT

<i>Event or Fault</i>	<i>Storage rules</i>	<i>Data to be recorded per event</i>
To be defined by Manufacturer	To be defined by Manufacturer	To be defined by Manufacturer

ANNEX 2

SEQUENCE DIAGRAMS OF MESSAGES EXCHANGES WITH THE ITS UNIT

Figure 118
Sequence Diagram for PIN validation attempt

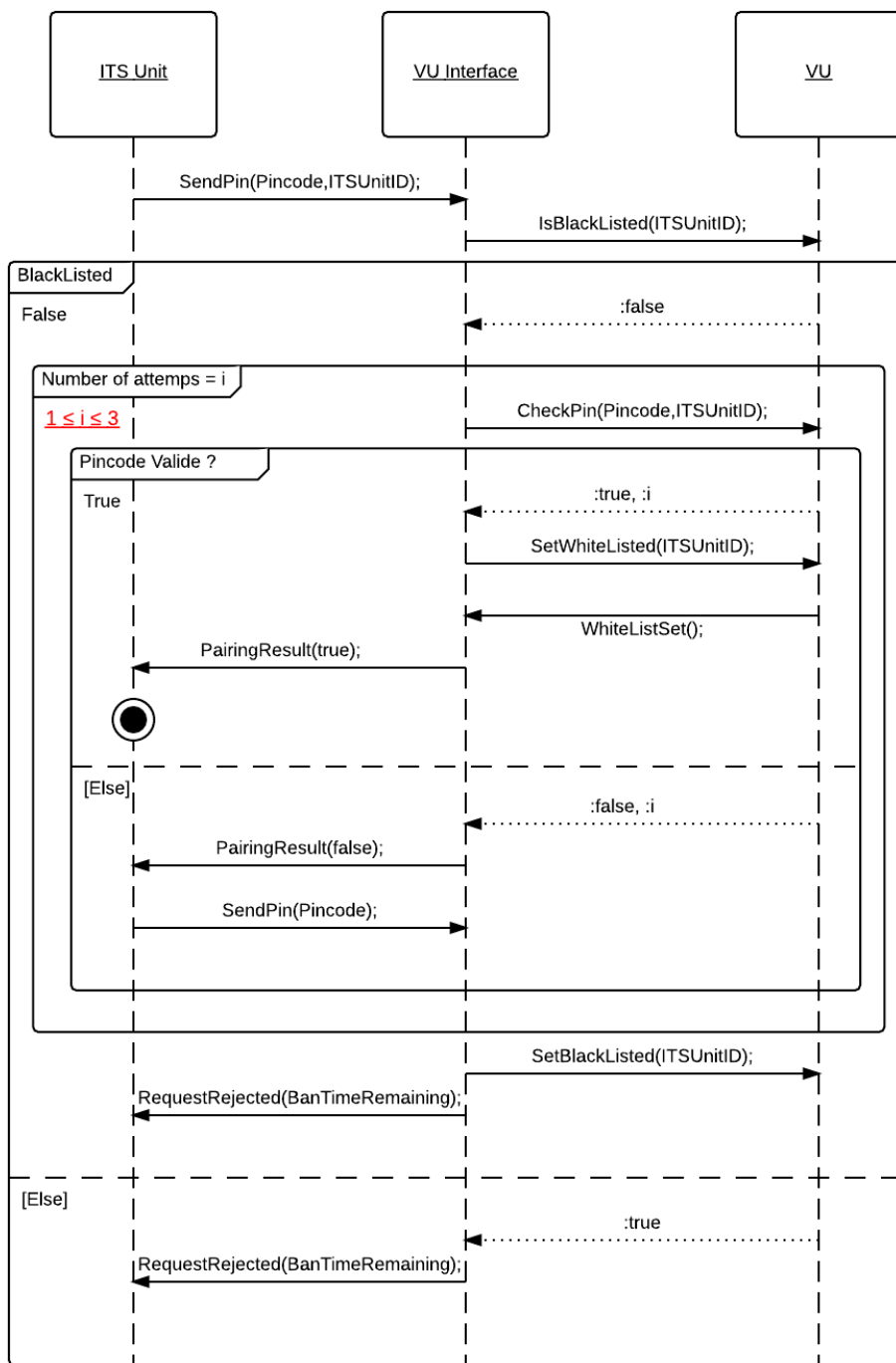


Figure 2-19
Sequence Diagram for ITS Unit's authorization verification

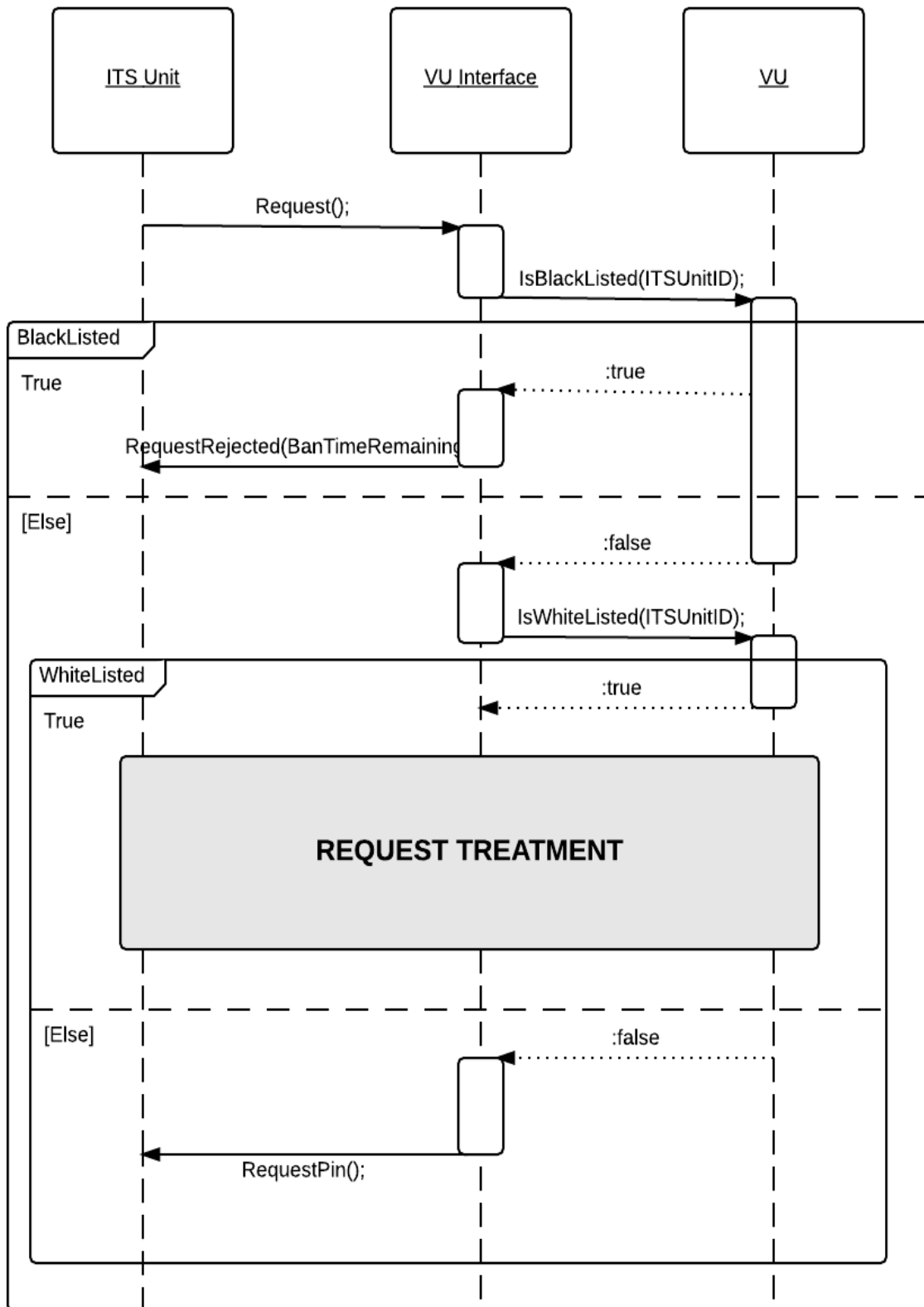


Figure 3 20
Sequence Diagram to process a request for data classified as non-personal (after correct PIN access)

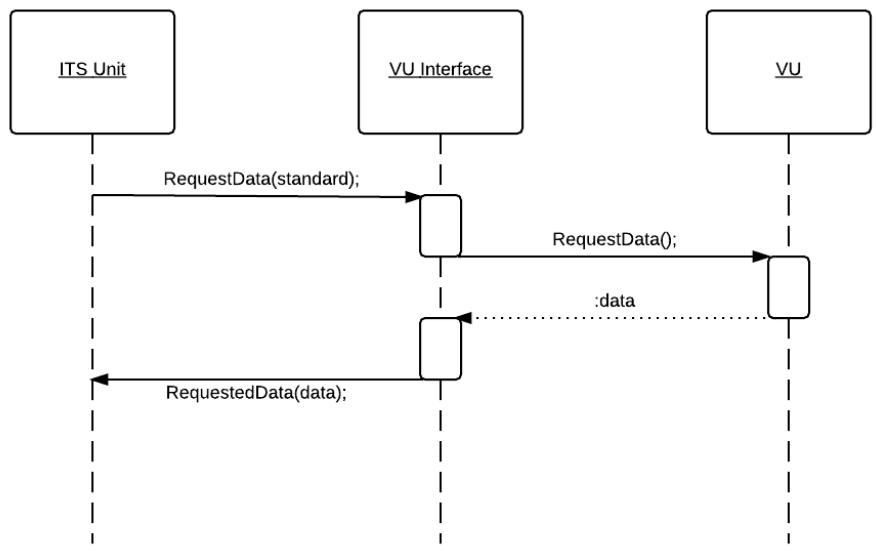


Figure 4 21
Sequence Diagram to process a request for data classified as personal (after correct PIN access)

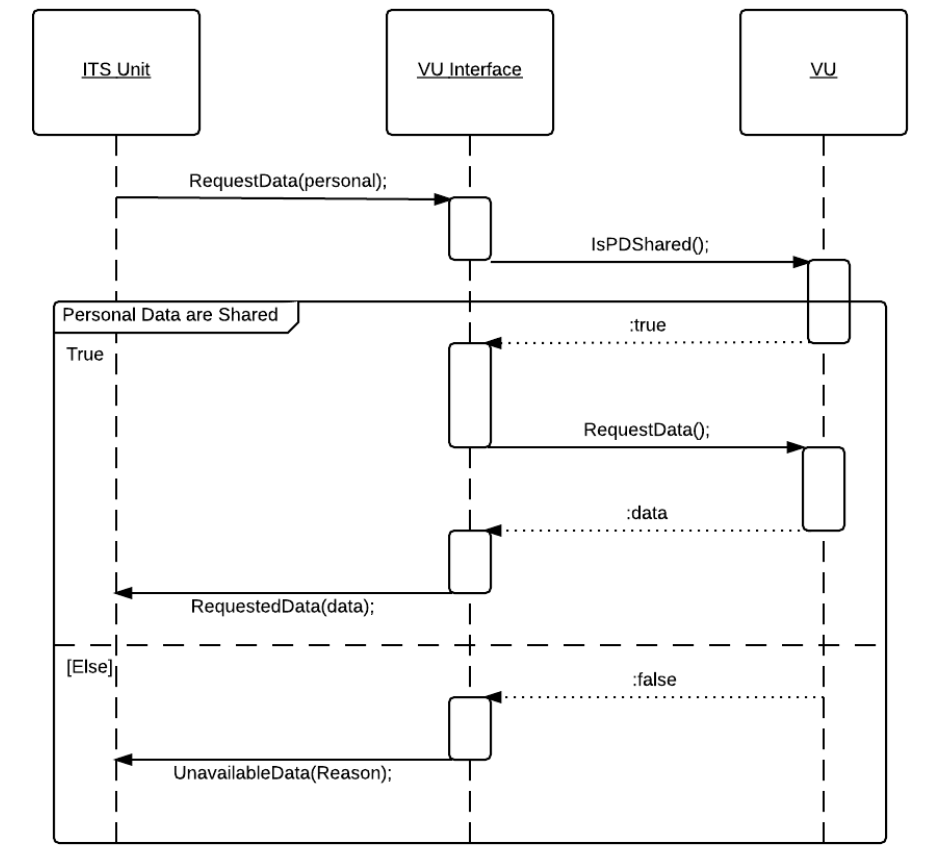
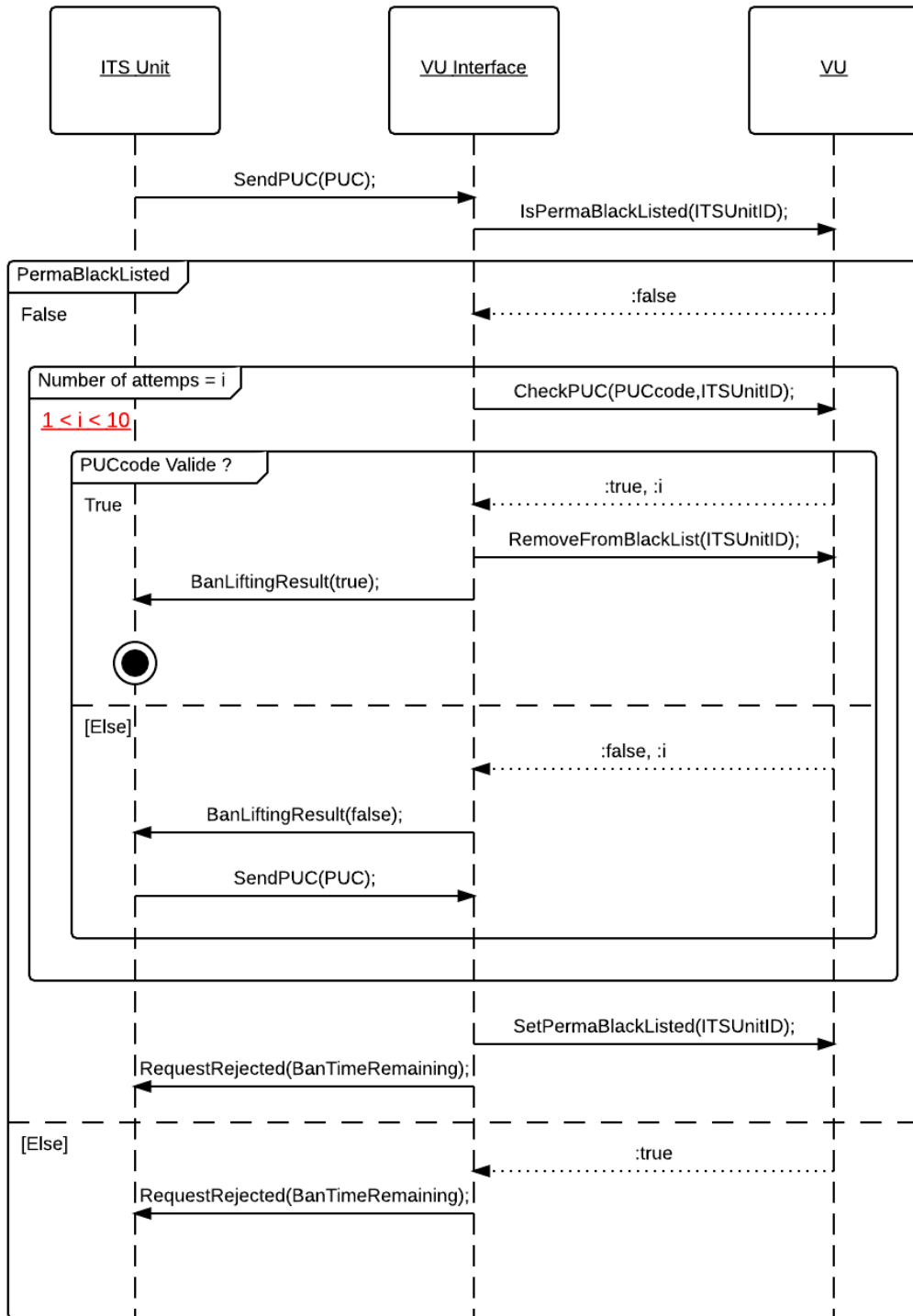


Figure 5 22
Sequence Diagram for PUC validation attempt



ANNEX 3

ASN.1 SPECIFICATIONS

```

FormatMessageModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
EXPORTS ;
IMPORTS SendPIN, SendPUC, PairingResult, RequestPIN, RequestRejected,
  BanLiftingResult FROM PINPUCDataFieldsModule
  RequestAccepted, RequestData, DataUnavailable FROM
  RequestDataFieldsModule
  SendITSID, NegativeAnswer FROM OtherDataFieldsModule;

-----
CompleteMessage ::= SEQUENCE{
  header Header,
  data DataField,
  checksum Checksum
}

-----
-----
-----
-----
-----
-----
-----
Header ::= SEQUENCE{
  tgt IDList,
  src IDList,
  len BIT STRING (1..255)
}

-----
vuID BIT STRING ::= 'EE'H
IDList ::= CHOICE{
  vu BIT STRING (vuID),
  itsUnits SEQUENCE OF BIT STRING,
    --Default hex Value:A0, redefined after first message exchange--
    --Each ID will be linked to the Bluetooth ID of the device--
  ...
}

-----
-----
-----
-----
-----
-----
-----
DATAFIELDS TYPES
-----
-----
DataField ::= SEQUENCE{
  sid BIT STRING,
  trtp BIT STRING,
  subMBytes SubMessageBytes,
  dataField Content,
  ...
}

-----
-----
-----
-----
-----
-----
-----
SubMessageBytes ::= SEQUENCE{
  currentSubM BIT STRING,
  totalSubM BIT STRING
}

```

```
Content ::= CHOICE{  
  requestPIN RequestPIN,  
  sendITSID SendITSID,  
  sendPin SendPIN,  
  pairRsIt PairingResult,  
  sendPUC SendPUC,  
  banlift BanLiftingResult,  
  requestRejected RequestRejected,  
  requestData RequestData,  
  requestOK RequestAccepted,  
  dataUnavailable DataUnavailable,  
  negAns NegativeAnswer  
}
```

~~CHECKSUM TYPES~~

```
Checksum ::= SEQUENCE{  
  SHA2 checksum  
}
```

END

```

PINPUCDataFieldsModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
EXPORTS SendPIN, SendPUC, PairingResult, RequestPIN, RequestRejected,
BanLiftingResult;
IMPORTS ;

-----
-----Utils-----
-----
-----
PUC ::= SEQUENCE (SIZE(8)) OF
INTEGER (SIZE(0..9))
-----
PIN ::= SEQUENCE (SIZE(4)) OF
INTEGER (SIZE(0..9))
-----
-----Messages From ITS Unit-----
-----
SendPIN {PIN:pin} ::= SEQUENCE {
sid BIT STRING ('03'H),
pin PIN (pin)
}
-----
SendPUC {PUC:puc} ::= SEQUENCE {
sid BIT STRING ('05'H),
puc PUC (puc)
}
-----
-----Messages From VU-----
-----
PairingResult ::= SEQUENCE{
sid BIT STRING ('04'H),
result BOOLEAN
}
-----
RequestPIN {MType:receivedRequest} ::= SEQUENCE{
sid BIT STRING ('01'H)
}
-----
RequestRejected ::= SEQUENCE{
sid BIT STRING ('07'H),
banTimeRemaining GeneralizedTime, PermaBan == 1k years }
-----
BanLiftingResult ::= SEQUENCE{
sid BIT STRING ('06'H),
result BOOLEAN
}
END

```

```

RequestDataFields DEFINITIONS AUTOMATIC TAGS ::= BEGIN
EXPORTS RequestAccepted, RequestData, DataUnavailable;
IMPORTS StandardEvent, PersonalEvent, StandardFault FROM EventsModule;

-----
From ITS Unit
-----

RequestData ::= SEQUENCE{
sid BIT STRING ('08'H),
requestedData DataTypeCode,
...
}

-----
From VU
-----

RequestAccepted ::= SEQUENCE{
sid BIT STRING ('09'H),
trtp DataTypeCode,
dataSheet CHOICE{
standardData StandardTachDataContent,
personalData PersonalTachDataContent,
gnss GNSSDataContent,
standardEvent StandardEventContent,
personalEvent PersonalEventContent,
standardFault StandardFaultContent,
manufacturerdata ManufacturerDataContent,
...
}
}

DataTypeCode ::= CHOICE{
standardTachData BIT STRING ('01'H),
personalTachData BIT STRING ('02'H),
gnssData BIT STRING ('03'H),
standardEventData BIT STRING ('04'H),
personalEventData BIT STRING ('05'H),
standardFaultData BIT STRING ('06'H),
manufacturerData BIT STRING ('07'H),
...
}

DataUnavailable ::= SEQUENCE{
sid BIT STRING ('0A'H),
trtp DataTypeCode,
reason UnavailableDataCodes
}

UnavailableDataCodes ::= CHOICE{
noDataAvailable BIT STRING ('10'H),
personalDataNotShared BIT STRING ('11'H),
...
}

```

~~Complete Tachograph Data~~

~~The format of the data was taken from the ISO16844-7 norm, more information available in this ISO document~~

~~Time ::= SEQUENCE{~~

~~seconds INTEGER (0..59.75), increment: 0.25s~~
~~minutes INTEGER (0..59), increment: 1min~~
~~hours INTEGER (0..23), increment: 1h~~
~~day INTEGER (0.25.. 31.75), increment: 0.25d~~
~~month INTEGER (1..12), increment: 1month~~
~~year INTEGER (1985..2235), increment: 1year~~
~~locMinOffset INTEGER (- 59..59), increment: 1min~~
~~locHourOffset INTEGER (- 23..23) increment: 1h~~

~~}~~

~~Date ::= SEQUENCE{~~

~~month INTEGER (1..12), increment: 1month~~
~~day INTEGER (0.25.. 31.75), increment: 0.25d~~
~~year INTEGER (1985..2235) increment: 1year~~

~~}~~

~~DriverName ::=SEQUENCE{~~

~~codePageSurname UTF8String, See ISO/IEC 8859~~
~~surname UTF8String,~~
~~codePageFirstname UTF8String, See ISO/IEC 8859~~
~~firstname UTF8String,~~

~~}~~

~~Message Content~~

~~StandardTachDataContent ::= SEQUENCE{~~

~~trtp DataTypeCode (DataTypeCode.&standardTachData),~~
~~personal BOOLEAN (FALSE),~~
~~data StandardTachyDataSheet,~~

~~}~~

~~PersonalTachDataContent ::= SEQUENCE{~~

~~trtp DataTypeCode (DataTypeCode.&personalTachData),~~
~~personal BOOLEAN (TRUE),~~
~~data PersonalTachyDataSheet~~

~~}~~

~~GNSSDataContent ::= SEQUENCE{~~

~~trtp DataTypeCode (DataTypeCode.&gnssData),~~
~~personal BOOLEAN (TRUE),~~
~~data GNSSDataSheet~~

~~}~~

~~StandardEventContent ::= SEQUENCE{~~

~~trtp DataTypeCode (DataTypeCode.&standardEventData),~~
~~personal BOOLEAN (FALSE),~~

```

----- data StandardEventDataSheet
----- }
-----
----- PersonalEventContent ::= SEQUENCE{
-----   trtp DataTypeCode (DataTypeCode.&personalEventData),
-----   personal BOOLEAN (TRUE),
-----   data PersonalEventDataSheet
----- }
-----
----- StandardFaultContent ::= SEQUENCE{
-----   trtp DataTypeCode (DataTypeCode.&standardFaultData),
-----   personal BOOLEAN (FALSE),
-----   data StandardFault
----- }
-----
----- ManufacturerDataContent ::= SEQUENCE{
-----   trtp DataTypeCode (DataTypeCode.&manufacturerData),
-----   personal BOOLEAN (TRUE),
-----   ...
----- }
-----
----- DATA SHEETS -----
-----
----- Data sheet format follows ISO 16844-7. -----
----- StandardTachyDataSheet ::= SEQUENCE{
-----   vin UTF8String (SIZE(17)),
-----   calibrationDate Date,
-----   driveRecognize INTEGER (2 UNION 12),
-----   driverCardDriver1 INTEGER (2 UNION 12),
-----   driverCardDriver2 INTEGER (2 UNION 12),
-----   timeDate Time,
-----   highResolutionTotalVehicleDistance INTEGER (0..21055406), increment: 5m
-----   serviceComponentIdentification INTEGER (0..255),
-----   serviceDelayCalendarTimeBased INTEGER ( 125..125), increment: 1week
-----   nextCalibrationDate Date,
-----   speedAuthorised INTEGER (0..250.996), increment 1/256km/h
-----   tachographCardsSlot1 INTEGER (0..4...), Maximum 250
-----   tachographCardsSlot2 INTEGER (0..4...), Maximum 250
-----   outOfScopeCondition INTEGER(2 UNION 12),
-----   modeOfOperation INTEGER (0..4...), Maximum 250
-----   registeringMemberState UTF8String, vehicleRegistrationNumber SEQUENCE {
-----     codePageVRN INTEGER (0..255),
-----     vrn OCTET STRING (SIZE(13)),
-----   },
-----   tachographNextMandatoryDownloadDate Date,
-----   ...
----- }
-----
----- PersonalTachyDataSheet ::= SEQUENCE{
-----   tachographVehicleSpeed INTEGER (0..250.996), increment 1/256km/h
-----   driver1WorkingState INTEGER (2 UNION 12 UNION 102 UNION 112 UNION 1002 UNION
1012...),

```



```


driver2WorkingState INTEGER (2 UNION 12 UNION 102 UNION 112 UNION 1002 UNION
1012...),
driver1TimeRelatedStates INTEGER(2 UNION 12 UNION 102 UNION 112 UNION 1002
UNION
1012 UNION 1102 UNION 1112 UNION 10002 UNION
10012 UNION
10102 UNION 10112 UNION 11002 UNION
11012...),
driver2TimeRelatedStates INTEGER(2 UNION 12 UNION 102 UNION 112 UNION 1002
UNION
1012 UNION 1102 UNION 1112 UNION 10002 UNION
10012 UNION
10102 UNION 10112 UNION 11002 UNION
11012...),
overSpeed INTEGER (2 UNION 12),
driver1Identification INTEGER (SIZE(19)), -- TODO NEED FURTHER SPECS FROM TACHO
REGULATION--
driver2Identification INTEGER (SIZE(19)), -- TODO NEED FURTHER SPECS FROM TACHO
REGULATION--
driver1ContinuousDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver2ContinuousDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver1CurrentDurationOfSelectedActivity INTEGER (0.. 64255), -- increment:
1min--
driver2CurrentDurationOfSelectedActivity INTEGER (0.. 64255), -- increment:
1min--
driver1Name DriverName,
driver2Name DriverName,
driver1CumulatedDrivingTimePreviousAndCurrentWeek INTEGER (0.. 64255), --
increment: 1min--
driver2CumulatedDrivingTimePreviousAndCurrentWeek INTEGER (0.. 64255), --
increment: 1min--
engineSpeed INTEGER(0..8031.875), -- increment: 0,125r/min--
driver1EndOfLastDailyRestPeriod Time,
driver2EndOfLastDailyRestPeriod Time,
driver1EndOfLastWeeklyRestPeriod Time,
driver2EndOfLastWeeklyRestPeriod Time,
driver1EndOfSecondLastWeeklyRestPeriod Time,
driver2EndOfSecondLastWeeklyRestPeriod Time,
driver1CurrentDailyDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver2CurrentDailyDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver1CurrentWeeklyDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver2CurrentWeeklyDrivingTime INTEGER (0.. 64255), -- increment: 1min--
driver1TimeLeftUntilNewDailyRestPeriod INTEGER (0.. 64255), -- increment: 1min--
-
driver2TimeLeftUntilNewDailyRestPeriod INTEGER (0.. 64255), -- increment: 1min--
-
driver1CardExpiryDate Date,
driver2CardExpiryDate Date,
driver1CardNextMandatoryDownloadDate Date,
driver2CardNextMandatoryDownloadDate Date,
driver1TimeLeftUntilNewWeeklyRestPeriod INTEGER (0.. 64255), -- increment:
1min--


```

```


driver2TimeLeftUntilNewWeeklyRestPeriod INTEGER (0.. 64255), --increment:
1min--
driver1NumberOfTimes9hDailyDrivingTimesExceeded INTEGER (0..13),
driver2NumberOfTimes9hDailyDrivingTimesExceeded INTEGER (0..13),
driver1CumulativeUninterruptedRestTime INTEGER (0.. 64255), --increment: 1min--
-
driver2CumulativeUninterruptedRestTime INTEGER (0.. 64255), --increment: 1min--
-
driver1MinimumDailyRest INTEGER (0.. 64255), --increment: 1min--
driver2MinimumDailyRest INTEGER (0.. 64255), --increment: 1min--
driver1MinimumWeeklyRest INTEGER (0.. 64255), --increment: 1min--
driver2MinimumWeeklyRest INTEGER (0.. 64255), --increment: 1min--
driver1MaximumDailyPeriod INTEGER (0..250), --increment: 1h--
driver2MaximumDailyPeriod INTEGER (0..250), --increment: 1h--
driver1MaximumDailyDrivingTime INTEGER (910 UNION 1010),
driver2MaximumDailyDrivingTime INTEGER (910 UNION 1010),
driver1NumberOfUsedReducedDailyRestPeriods INTEGER (0..13),
driver2NumberOfUsedReducedDailyRestPeriods INTEGER (0..13),
driver1RemainingCurrentDrivingTime INTEGER (0.. 64255), --increment: 1min--
driver2RemainingCurrentDrivingTime INTEGER (0.. 64255), --increment: 1min--
...
}
}

GNSSDataSheet ::= SEQUENCE {
gnssPosition GeoCoordinates
-- See Appendix 1 for definition of GeoCoordinates --
}

StandardEventDataSheet ::= SEQUENCE{
events SEQUENCE OF StandardEvent
}

PersonalEventDataSheet ::= SEQUENCE{
events SEQUENCE OF PersonalEvent
}
}
END

EventsModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
EXPORTS ALL;
IMPORTS NationAlpha FROM Annex Appendix1; -- See Annex Appendix 1 for more information
about NationAlpha --

SecurityBreachEvent ::= SEQUENCE{
-- See Appendix 1B for more information --
}

RecordingEquipmentFaultType ::= SEQUENCE{
-- See Appendix 1B for more information --
}

StandardEvent ::= CHOICE{
insertionInvalidCard InsertionOfANonValidCard,
cardConflict CardConflict,
timeOverlap TimeOverlap,


```

```

previousSessionNotClosed LastCardSessionNotCorrectlyClosed,
overSpeeding OverSpeeding,
powerSupplyInterruption PowerSupplyInterruption,
comErrorWithRemoteFacility
CommunicationErrorWithTheRemoteCommunicationFacility,
absenceGNSSPosition AbsenceOfPositionInformationFromGNSSReceiver,
positionDataError PositionDataError,
motionDataError MotionDataError,
vehicleMotionConflict VehicleMotionConflict,
securityBreachAttempt SecurityBreachAttempt,
timeConflict TimeConflict,
...
}
PersonalEvent ::= CHOICE{
  lackOfAppropriateCard DrivingWithoutAnAppropriateCard,
  cardInsertionWhileDriving CardInsertionWhileDriving,
  overSpeeding OverSpeeding,
  ...
}
StandardFault ::= CHOICE{
  cardFault CardFault,
  recordingEquipmentFault RecordingEquipmentFault,
  ...
}
-----
EVENTS LIST
-----
InsertionOfANonValidCard ::= SEQUENCE{
  beginDate GeneralizedTime,
  endDate GeneralizedTime,
  carsdType SEQUENCE OF UTF8String,
  cardsNumber SEQUENCE OF INTEGER,
  issuingMemberState SEQUENCE OF NationAlpha,
  cardsGeneration SEQUENCE OF INTEGER
}
CardConflict ::= SEQUENCE{
  beginDate GeneralizedTime,
  endDate GeneralizedTime,
  carsdType SEQUENCE OF UTF8String,
  cardsNumber SEQUENCE OF INTEGER,
  issuingMemberState SEQUENCE OF NationAlpha,
  cardsGeneration SEQUENCE OF INTEGER
}
TimeOverlap ::= SEQUENCE{
  beginDate GeneralizedTime,
  endDate GeneralizedTime,
  carsdType SEQUENCE OF UTF8String,
  cardsNumber SEQUENCE OF INTEGER,

```

```

_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ numberSimilarEvent INTEGER
_____ }
_____
_____ DrivingWithoutAnAppropriateCard ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ numberOfSimilarEvent INTEGER
_____ }
_____
_____ CardInsertionWhileDriving ::= SEQUENCE{
_____ date GeneralizedTime,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ numberOfSimilarEvents INTEGER
_____ }
_____
_____ LastCardSessionNotCorrectlyClosed ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ oldSession SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ vrn UTF8String,
_____ issuingMemberState NationAlpha,
_____ cardsGeneration INTEGER,
_____ }
_____ }
_____ }
_____
_____ OverSpeeding ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ maximumSpeed INTEGER,
_____ averageSpeed INTEGER,
_____ cardType UTF8String,
_____ cardNumber INTEGER,
_____ issuingMemberState NationAlpha,
_____ cardGeneration INTEGER,
_____ numberOfSimilarEvents INTEGER
_____ }
_____
_____ PowerSupplyInterruption ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,

```

```

----- cardsType SEQUENCE OF UTF8String,
----- cardsNumber SEQUENCE OF INTEGER,
----- issuingMemberState SEQUENCE OF NationAlpha,
----- cardsGeneration SEQUENCE OF INTEGER,
----- numberOfSimilarEvent INTEGER
----- }
-----
----- CommunicationErrorWithTheRemoteCommunicationFacility ::= SEQUENCE{
-----     beginDate GeneralizedTime,
-----     endDate GeneralizedTime,
-----     cardsType SEQUENCE OF UTF8String,
-----     cardsNumber SEQUENCE OF INTEGER,
-----     issuingMemberState SEQUENCE OF NationAlpha,
-----     cardsGeneration SEQUENCE OF INTEGER,
-----     numberOfSimilarEvent INTEGER
----- }
-----
----- AbsenceOfPositionInformationFromGNSSReceiver ::= SEQUENCE{
-----     beginDate GeneralizedTime,
-----     endDate GeneralizedTime,
-----     cardsType SEQUENCE OF UTF8String,
-----     cardsNumber SEQUENCE OF INTEGER,
-----     issuingMemberState SEQUENCE OF NationAlpha,
-----     cardsGeneration SEQUENCE OF INTEGER,
-----     numberOfSimilarEvent INTEGER
----- }
-----
----- PositionDataError ::= SEQUENCE{
-----     beginDate GeneralizedTime,
-----     endDate GeneralizedTime,
-----     cardsType SEQUENCE OF UTF8String,
-----     cardsNumber SEQUENCE OF INTEGER,
-----     issuingMemberState SEQUENCE OF NationAlpha,
-----     cardsGeneration SEQUENCE OF INTEGER,
-----     numberOfSimilarEvent INTEGER
----- }
-----
----- MotionDataError ::= SEQUENCE{
-----     beginDate GeneralizedTime,
-----     endDate GeneralizedTime,
-----     cardsType SEQUENCE OF UTF8String,
-----     cardsNumber SEQUENCE OF INTEGER,
-----     issuingMemberState SEQUENCE OF NationAlpha,
-----     cardsGeneration SEQUENCE OF INTEGER,
-----     numberOfSimilarEvent INTEGER
----- }
-----
----- VehicleMotionConflict ::= SEQUENCE{
-----     beginDate GeneralizedTime,
-----     endDate GeneralizedTime,
-----     cardsType SEQUENCE OF UTF8String,
-----     cardsNumber SEQUENCE OF INTEGER,
-----     issuingMemberState SEQUENCE OF NationAlpha,
-----     cardsGeneration SEQUENCE OF INTEGER,

```

```

_____ numberOfSimilarEvent INTEGER
_____ }
_____

SecurityBreachAttempt ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ numberOfSimilarEvent INTEGER,
_____ }

SecurityBreachAttempt ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime OPTIONAL,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ numberOfSimilarEvent INTEGER,
_____ typeOfEvent SecurityBreachEvent
_____ }

_____

TimeConflict ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ numberOfSimilarEvent INTEGER
_____ }

_____

-----
----- FAULTS LIST
-----

CardFault ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ cardsGeneration SEQUENCE OF INTEGER,
_____ }

RecordingEquipmentFault ::= SEQUENCE{
_____ beginDate GeneralizedTime,
_____ endDate GeneralizedTime,
_____ faultType RecordingEquipmentFaultType,
_____ cardsType SEQUENCE OF UTF8String,
_____ cardsNumber SEQUENCE OF INTEGER,
_____ issuingMemberState SEQUENCE OF NationAlpha,
_____ }

```

cardsGeneration SEQUENCE OF INTEGER,
}
END

Sub-appendix 14. Remote communication function

TABLE OF CONTENT

1.	INTRODUCTION	554
2.	SCOPE	555
3.	ACRONYMS, DEFINITIONS AND NOTATIONS	556
4.	OPERATIONAL SCENARIOS	558
4.1.	Overview	558
4.1.1	Preconditions to data transfer via 5.8 GHz DSRC interface	558
4.1.2	Profile 1a: via a hand aimed or temporary roadside mounted and aimed Remote Early Detection Communication Reader	558
4.1.3	Profile 1b: via a vehicle mounted and directed Remote Early Detection Communication Reader (REDCR).....	558
4.2.	Security/Integrity	558
5.	REMOTE COMMUNICATION DESIGN AND PROTOCOLS.....	558
5.1.	Design	560
5.2.	Workflow	564
5.2.1	Operations	564
5.2.2	Interpretation of the Data received via the DSRC communication	565
5.3.	DSRC Physical interface parameters for remote communication	565
5.3.1	Location constraints	565
5.3.2	Downlink and uplink parameters.....	465
5.3.3	Antenna design	465
5.4.	DSRC Protocol requirements for RTM	570
5.4.1	Overview	570
5.4.2	Commands	570
5.4.3	Interrogation command sequence	573
5.4.4	Data structures.....	573
5.4.5	Elements of RtmData, actions performed and definitions.....	574
5.4.6	Data transfer mechanism	576
5.4.7	Detailed DSRC transaction description	584
5.4.8	DSRC Test transaction description	584
5.5.	Reserved for future use Support for Directive (EU) 2015/719.....	593
5.5.1	Overview	595
5.5.2	Commands	595
5.5.3	Interrogation command sequence.....	595
5.5.4	Data structures.....	595
5.5.5	ASN.1 module for the OWS DSRC transaction.....	595
5.5.6	Elements of OwsData, actions performed and definitions	597
5.5.7	Data transfer mechanisms.....	597
5.6.	Data transfer between the DSRC-VU and VU.....	595
5.6.1	Physical Connection and interfaces.....	596
5.6.2	Application Protocol.....	597
5.7.	Error handling	597
5.7.1	Recording and communication of the Data in the DSRC VU.....	597
5.7.2	Wireless Communication errors.....	598

6.	COMMISSIONING AND PERIODIC INSPECTION TESTS FOR THE REMOTE COMMUNICATION FUNCTION ..	599
6.1.	<i>General</i>	<i>599</i>
6.2.	<i>ECHO</i>	<i>599</i>
6.3.	<i>Tests to validate the secure data content.....</i>	<i>600</i>
<i>annex</i>	<i>.....</i>	<i>601</i>

1. Introduction

This ~~Appendix~~ **sub-appendix** specifies the design and the procedures to follow in order to perform the remote communication function (the Communication) ~~as required in Article 9 of Regulation (EU) No 165/2014 (the Regulation).~~

~~DSC_1 The Regulation (EU) No 165/2014 determines that the~~ **DSC_1** The tachograph shall be equipped with a remote communication functionality that shall enable agents of the competent control authorities to read tachograph information from passing vehicles by using remote interrogation equipment (the Remote early detection communication reader [REDCR]), specifically, interrogation equipment connecting wirelessly using CEN 5.8 GHz Dedicated Short Range Communication (DSRC) interfaces.

It is important to comprehend that this functionality is intended to serve only as a pre-filter in order to select vehicles for closer inspection, and it does not replace the formal inspection process ~~as determined in the provisions of Regulation (EU) n. 165/2014. See recital 9 in the preamble of this regulation stating that remote-~~ **Remote** communication between the tachograph and control authorities for roadside control purposes facilitates targeted roadside checks.

DSC_2 The Data shall be exchanged using the Communication which shall be a wireless intercourse using 5.8 GHz DSRC wireless communications consistent with this ~~Appendix~~ **sub-appendix** and tested against the appropriate parameters of EN 300 674-1, {Electromagnetic compatibility and Radio spectrum Matters (ERM); Road Transport and Traffic Telematics (RTTT); Dedicated Short Range Communication (DSRC) transmission equipment (500 kbit/s / 250 kbit/s) operating in the 5,8 GHz Industrial, Scientific and Medical (ISM) band; Part 1: General characteristics and test methods for Road Side Units (RSU) and On -Board Units (OBU)}.

DSC_3 The Communication shall be established with the communications equipment only when so requested by the equipment of the competent control authority using compliant radio-communication means (the Remote early detection communication reader (REDCR)).

DSC_4 The Data shall be secured to ensure integrity.

DSC_5 Access to the Data communicated shall be restricted to competent control authorities ~~authorised to check infringement of Regulation (EC) no. 561/2016 and of Regulation (EU) no. 165/2014~~ and to workshops in so far as it is necessary to verify the correct functioning of the tachograph.

DSC_6 The Data exchanged during the Communication shall be limited to the data necessary for the purpose of targeting roadside checks of vehicles with a potentially manipulated or misused tachograph.

DSC_7 Data integrity and security shall be obtained by securing the Data within the Vehicle Unit (VU) and by passing only the secured payload data and security related data (see 4.4.4 **5.4.4**) across the wireless 5.8 GHz DSRC remote communication medium, meaning that only authorised persons of competent control authorities have the means to understand the data passed across the Communication and to verify its authenticity. See ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms.

DSC_8 *The Data* shall contain a timestamp for the time of its last update.

DSC_9 The content of the security data shall be known only to and within the control of the competent control authorities, and those parties with whom they share this information and is out with the provisions of *the Communication* that is the subject of this ~~Appendix~~ **sub-appendix**, save that *the Communication* makes provision to transfer a packet of security data with every packet of payload data.

DSC_10 The same architecture and equipment shall be capable be used to acquire other data concepts (such as weigh-on-board) using the architecture specified herein.

DSC_11 ~~For clarification in accordance with the provisions of Regulation (EU) No 165/2014 (Article 7) data, Data~~ concerning the identity of the driver shall not be communicated across *the Communication*.

2. Scope

The scope of this ~~Appendix~~ **sub-appendix** is to specify how agents of the competent control authorities use a specified 5.8 GHz DSRC wireless communication to remotely obtain data (*the Data*) from a targeted vehicle that identifies that the targeted vehicle is in potential violation of ~~Regulation (EU) No. 165/2014~~ **this Agreement** and should be targeted for consideration to be stopped for further investigation.

~~The Regulation (EU) No. 165/2014~~ Data collected shall be limited to data or pertaining to data that identifies a potential infringement ~~as defined in Article 9 of Regulation (EU) No. 165/2014~~.

In this scenario, the time available for communication is limited, because *the Communication* is targeted and of a short- range design. Further, the same communication means for remote tachograph monitoring (RTM) may also be used by the competent control authorities for other applications (such as the maximal weights and dimensions for heavy goods vehicles ~~defined in Directive 2015/719/EC~~) and such operations may be separate or sequential at the discretion of the competent control authorities.

This ~~Appendix~~ **sub-appendix** specifies:

- The communications equipment, procedures and protocols to be used for *the Communication*
- The Standards and Regulations to which the radio equipment shall comply
- The presentation of *the Data* to *the Communication* equipment
- The enquiry and download procedures and sequence of operations
- The Data to be transferred
- Potential interpretation of *the Data* transferred across *the Communication*
- The provisions for security data relating to *the Communication*
- The availability of *the Data* to the competent control authorities
- How the *Remote early detection communication reader* can request different freight and fleet data concepts

For clarification, this ~~Appendix~~ **sub-appendix** does not specify:

the collection of *the Data* operation and management within the VU (~~which shall be a function of product design unless specified elsewhere within Regulation (EU) No. 165/2014~~)

- the form of presentation of collected data to the agent of the competent control authorities, nor the criteria which shall be used by the competent control authorities to decide which vehicles to stop (~~which shall be a function of product design unless specified elsewhere within Regulation (EU) No. 165/2014 or a policy decision of the competent control authorities~~). For clarification: *the Communication* only makes *the Data* available to the competent control authorities in order that they may make informed decisions

- Data security provisions (such as encryption) concerning the content of *the Data* (which shall be specified within ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms).
- detail of any data concepts other than RTM which may be obtained using the same architecture and equipment
- detail of the behaviour and management between VU's and the DSRC-VU, nor the behaviour within the DSRC-VU (other than to provide *the Data* when so requested by an REDCR).

3. Acronyms, definitions and notations

The following acronyms and definitions specific to this ~~Appendix~~ **sub-appendix** are used in this **sub**-appendix:

the Antenna - electrical device which converts electric power into radio waves, and vice versa used in combination with a radio transmitter or radio receiver. In operation, a radio transmitter supplies an electric current oscillating at radio frequency to the antenna's terminals, and the antenna radiates the energy from the current as electromagnetic waves (radio waves). In reception, an antenna intercepts some of the power of an electromagnetic wave in order to produce a tiny voltage at its terminals, that is applied to a receiver to be amplified

the Communication - exchange of information/data between a DSRC-REDCR and a DSRC-VU according to section 0 in a master-slave relationship to obtain the Data.

the Data - secured data of defined format (see **5.4.4**) requested by the DSRC-REDCR and provided to the DSRC-REDCR by the DSRC-VU across a 5.8 GHz DSRC link as defined in 5 below

~~Regulation (EC) No. 165/2014 Regulation (EU no. 165/2014 of the European Parliament and the Council of 4 February 2014 on tachograph in road transport repealing Council Regulation (EEC) No. 3821/85 on recording equipment in road transport and amending Regulation (EC) No 561/2016 of the European Parliament — and the Council on the harmonisation of certain social legislation relating to road transport.~~

AID	Application Identifier
BLE	Bluetooth Low Energy
BST	Beacon Service Table
CIWD	Card insertion while driving
CRC	cyclic redundancy check
DSC (n)	identifier of a requirement for a specific DSRC appendix
DSRC	Dedicated Short Range Communication
DSRC-REDCR	DSRC – Remote Early Detection Communication Reader.
DSRC-VU	DSRC – Vehicle Unit. This is the “remote early detection facility” defined in Annex Appendix 1C.
DWVC	Driving without valid card
EID	Element Identifier

LLC	Logical Link Control
LPDU	LLC Protocol Data Unit
OWS	Onboard Weighing System
PDU	Protocol Data Unit
REDCR	Remote early detection communication reader. This is the “remote early detection communication reader equipment” defined in Annex Appendix 1C .
RTM	Remote Tachograph Monitoring
SM-REDCR	Security Module-Remote early detection communication reader
TARV of	Telematics Applications for Regulated Vehicles (ISO 15638 series Standards)
VU	Vehicle Unit
VUPM	Vehicle Unit Payload Memory
VUSM	Vehicle Unit Security Module
VST	Vehicle Service Table
WIM	Weigh in motion
WOB	Weigh on board

The specification defined in this ~~Appendix~~ **sub-appendix** refers to and depends upon all or parts of the following regulations and standards. Within the clauses of this ~~Appendix~~ **sub-appendix** the relevant standards, or relevant clauses of standards, are specified. In the event of any contradiction the clauses of this sub-appendix shall take precedence. In the event of any contradiction where no specification is clearly determined in this ~~Appendix~~ **sub-appendix**, operating within ERC 70-03 (and tested against the appropriate parameters of EN 300 674-1)¹⁵ shall take precedence, followed in descending order of preference by¹⁶ EN 12795, EN 12253 EN 12834 and EN 13372, 6.2, 6.3, 6.4 and 7.1.

~~Regulations and standards~~ **Standards** referenced in this ~~Appendix~~ **sub-appendix** are:

~~Regulation (EU) No. 165/2014 of the European Parliament and the Council of 4 February 2014 on tachograph in road transport repealing Council Regulation (EEC) No. 3821/85 on recording equipment in road transport and amending Regulation (EC) No 561/2016 of the European Parliament and of the Council of 15 March 2006 on the harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) No. 3821/85 and (EC) No. 21358 and repealing Council Regulation (EEC) No. 3820/85 (Text with EEA relevance).~~

1. ERC 70-03 CEPT: ECC Recommendation 70-03: Relating to the Use of Short Range Devices (SRD)
2. ISO 15638 Intelligent transport systems — Framework for cooperative telematics applications for regulated commercial freight vehicles (TARV).

¹⁵ Given the development of scientific and technological progress in cases where there is no reference to any ISO standard, but only a reference to a regional standard, the competent authorities of the Contracting Parties may use other technical rules, but no worse than those provided for in the applicable standards.

¹⁶ Conversion to ISO standard planned over a five-year period

3. EN 300 674-1 Electromagnetic compatibility and Radio spectrum Matters (ERM); Road Transport and Traffic Telematics (RTTT); Dedicated Short Range Communication (DSRC) transmission equipment (500 kbit/s / 250 kbit/s) operating in the 5,8 GHz Industrial, Scientific and Medical (ISM) band; Part 1: General characteristics and test methods for Road Side Units (RSU) and On-Board Units (OBU).
4. EN 12253 Road transport and traffic telematics - Dedicated short-range communication - Physical layer using microwave at 5.8 GHz.
5. EN 12795 Road transport and traffic telematics – Dedicated short-range communication - Data link layer: medium access and logical link control.
6. EN 12834 Road transport and traffic telematics – Dedicated short-range communication - Application layer.
7. EN 13372 Road transport and traffic telematics – Dedicated short-range communication – Profiles for RTTT applications
8. ISO 14906 Electronic fee collection – Application interface definition for dedicated short-range communication

4. Operational Scenarios

4.1. Overview

~~Regulation (EU) No. 165/2014 provides specific and controlled~~ **The supported** scenarios within which the Communication is to be used are:

“Communication Profile 1: Roadside inspection using a short range wireless communication Remote Early Detection Communication Reader instigating a physical roadside inspection (master--slave)

Reader Profile 1a: via a hand aimed or temporary roadside mounted and aimed Remote Early Detection Communication

Reader Profile 1b: via a vehicle mounted and directed Remote Early Detection Communication Reader”.

4.1.1 Preconditions to data transfer via 5.8 GHz DSRC interface

NOTE: In order to understand the context of the preconditions the reader is referred to Figure 14.3 below.

4.1.1.1 Data held in VU

DSC_12 The VU shall be responsible to keep updated every 60 seconds and maintain the data to be stored in the VU, without any involvement of the DSRC communication function. The means by which this is achieved is internal to the VU, specified in ~~Regulation (EU) No. 165/2014, Annex 1C~~ **Appendix 1C**, section 3.19 “~~Remote communication for targeted roadside checks~~” and is not specified in this ~~Appendix~~ **sub-appendix**.

4.1.1.2 Data provided to DSRC-VU Facility

DSC_13 The VU shall be responsible to update the DSRC tachograph data (*the Data*) whenever the data stored in the VU is updated at the interval determined in 4.1.1.1 (DSC_12), without any involvement of the DSRC communication function.

DSC_14 The VU data shall be used as a basis to populate and update *the Data*, the means by which this is achieved, is specified in ~~Annex 1C~~ **Appendix 1C**, section 3.19 “*Remote communication for targeted roadside checks*” or if there is no such specification it is a function of product design and is not specified in this ~~Appendix~~ **sub-appendix**. For the design of the connection between DSRC-VU facility and the VU, please refer to section 5.6.

4.1.1.3 Content of the Data

DSC_15 The content and format of *the Data* shall be such that, once decrypted, it shall be structured and made available in the form and format specified in 5.4.4 of this **sub-appendix** (Data structures).

4.1.1.4 Data presentation

DSC_16 *The Data*, having been kept frequently updated in accordance with the procedures determined in 4.1.1.1, shall be secured prior to presentation to the *DSRC-VU*, and presented as a secured data concept value, for temporary storage in the *DSRC-VU* as the current version of *the Data*. This data is transferred from the *VUSM* to the DSRC function *VUPM*. The *VUSM* and *VUPM* are functions and not necessarily physical entities. The form of physical instantiation to perform these functions shall be a matter of product design unless specified elsewhere in ~~Regulation (EU) No. 165/2014~~ **Appendix 1C**.

4.1.1.5 Security data

DSC_17 Security data (*DSRCsecurityData*), comprising the data required by the *REDCR* to complete its ability to decrypt *the Data* shall be supplied as defined in ~~Appendix~~ **Sub-appendix 11 Common Security Mechanisms** and presented as a data concept value, for temporary storage in the *DSRC-VU* as the current version of *DSRCsecurityData*, in the form defined in ~~section 5.4.4 of this Appendix-sub-appendix~~ **section 5.4.4**.

4.1.1.6 VUPM data available for transfer across the DSRC interface

DSC_18 The data concept which shall always be available in the DSRC function *VUPM* for immediate transfer upon request by the *REDCR* is defined in section 5.4.4 for full ASN.1 Module specifications.

General overview of communication Profile 1

This profile covers the use case where an agent of the competent control authorities, uses a short range remote communication Remote Early Detection Communication Reader (5.8 GHz DSRC interfaces operating within ERC 70-03, and tested against the appropriate parameters of EN 300 674-1 as described in section 0) (*the REDCR*) to remotely identify a vehicle which is potentially in violation of ~~Regulation (EU) No. 165/2014~~. Once identified, the agent of the competent control authorities who is controlling the interrogation decides whether the vehicle should be stopped.

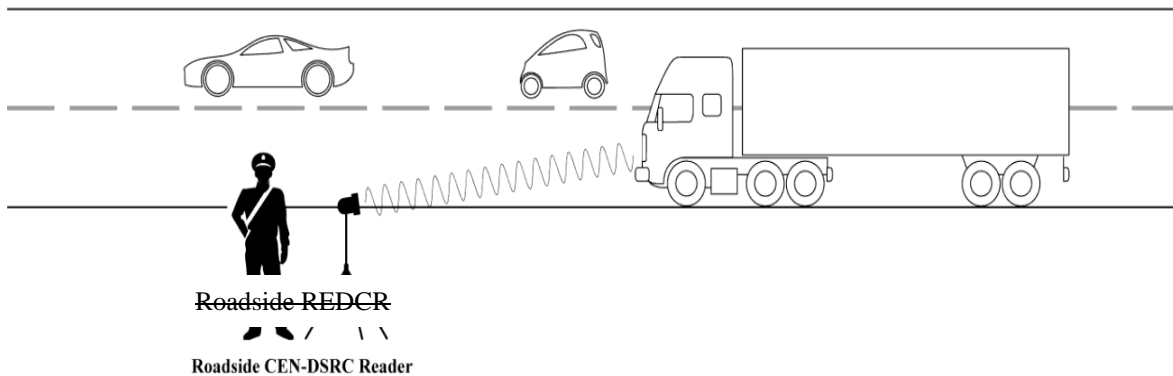
4.1.2 Profile 1a: via a hand aimed or temporary roadside mounted and aimed Remote Early Detection Communication Reader

In this use case the agent of the competent control authorities is situated at the roadside, and aims a hand held, tripod mounted, or similar portable, *REDCR* from the roadside towards the centre of the windshield of the targeted vehicle. The interrogation is made using 5.8 GHz DSRC interfaces operating within ERC 70-03, and tested against the appropriate parameters of EN 300 674-1 as described in section 0. See Figure 14.1 (Use Case 1).

Figure 14.1

Roadside interrogation using 5.8 GHz DSRC

Use case 1

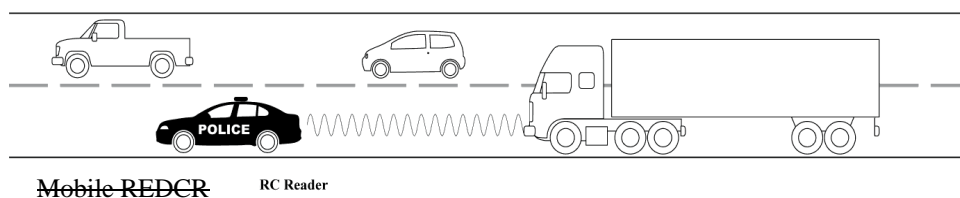


4.1.3. Profile 1b: via a vehicle mounted and directed Remote Early Detection Communication Reader (REDCR)

In this use case the agent of the competent control authorities is situated within a moving vehicle, and either aims a hand held, portable *REDCR* from the vehicle towards the centre of the windshield of the targeted vehicle, or *the REDCR* is mounted within or on the vehicle so as to point towards the centre of the windshield of the targeted vehicle when the Remote Early Detection Communication Reader’s vehicle is in a particular position relevant to the targeted vehicle (for example directly ahead in a stream of traffic). The interrogation is made using 5.8 GHz DSRC interfaces operating within ERC 70-03, and tested against the appropriate parameters of EN 300 674-1 as described in section 0. See Figure 14.2. (Use Case 2).

Figure 14.2
Vehicle based interrogation using 5.8 GHz DSRC

Use case 2



4.2. Security/Integrity

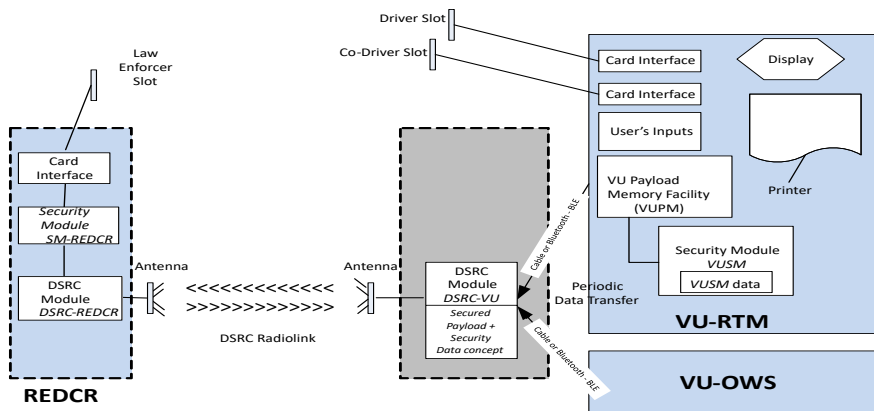
To give the possibility to verify the authenticity and integrity of downloaded data through the remote communication, the secured *Data* is verified and decrypted in accordance with ~~Appendix~~ **Sub-appendix 11** Common Security Mechanisms.

5. Remote Communication design and protocols

5.1. Design

The design of the remote communication function in the Smart Tachograph is shown as described in Figure 14.3.

Figure 14.3
Design of the remote communication function



DSC_19 The following functions are located in the VU:

- Security Module (*VUSM*). This function present in the VU is responsible for securing *the Data* which is to be transmitted from the *DSRC-VU* to the agent of the competent control authorities via remote communication.
- The secured data is stored in the *VUSM* memory. At intervals determined in 4.1.1.1 (DSC_12), the VU encrypts and replenishes the RTMdata concept (which comprises payload data and security data concept values determined below in this **Appendix sub-appendix**) held in the memory of the DSRC-VU. The operation of the security module is defined in **Appendix Sub-appendix 11 Common Security Mechanisms** and outwith the scope of this **Appendix sub-appendix**, save that it shall be required to provide updates to the VU Communication facility each time the *VUSM* data changes.
- The communication between the VU and the DSRC-VU may be a wired communication or a Bluetooth Low Energy (BLE) communication, and the physical location of the DSRC-VU may be integral with the antenna on the windshield of the vehicle, may be internal to the VU, or located somewhere between.
- The DSRC-VU shall have a reliable source of power available at all times. The means by which it is provided with its power is a design decision.
-
- The memory of the DSRC-VU shall be non-volatile in order to maintain the *Data* in the DSRC-VU even when the vehicle ignition is switched off.
- If the communication between the VU and the DSRC-VU is made via BLE and the power source is a non-recharging battery, the power source of the DSRC-VU shall be replaced at every Periodic Inspection, and the manufacturer of the DSRC-VU equipment shall be responsible to ensure that the power supply is adequate to last from one Periodic Inspection to the next Periodic Inspection, maintaining normal access to the data by an REDCR throughout the period without failure or interruption.
- VU RTM ‘payload memory’ facility (*VUPM*). This function present in the VU is responsible for providing and updating *the Data*. The content of *The Data*. (“TachographPayload”) is defined in 5.4.4/5.4.5 below and is updated at the interval determined in 4.1.1.1 (DSC_12).
- DSRC-VU. This is the function, within or connected to the antenna and in communication with the VU through a wired or wireless (BLE) connection, which

holds the current data (*VUPM-data*) and manages the response to an interrogation across the 5.8 GHz DSRC medium. Disconnection of the DSRC facility or interference during normal vehicle operation with the functioning of the DSRC facility shall be construed as a violation of ~~Regulation (EU) No. 165/2014~~ **this Agreement**.

- Security module (REDCR) (*SM-REDCR*) is the function used to decrypt and check integrity of the data originating from the VU. The means by which this is achieved is determined in ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms, and is not defined in this ~~Appendix~~ **sub-appendix**.
- The DSRC facility (REDCR) (*DSRC-REDCR*) function comprises a 5.8 GHz transceiver and associated firmware and software which manages *the Communication* with the *DSRC-VU* according to this ~~Appendix~~ **sub-appendix**.
- The *DSRC-REDCR* interrogates the *DSRC-VU* of the targeted vehicle and obtains *the Data* (the targeted vehicle's current *VUPM-data*) via the DSRC link and processes and stores the received data in its *SM-REDCR*.
- The DSRC-VU antenna shall be positioned at a location where it optimizes the DSRC communication between the vehicle and the roadside **reader** antenna (~~in general in or close to, when the reader is installed 15 meters distance in front of the vehicle and 2 meters high, targeting the horizontal and vertical centre of the vehicle windshield ...~~) **windscreen**. For light vehicles an installation **corresponding** to the upper part of the windscreen is suitable. **For all the other vehicles the DSRC antenna shall be installed either near the lower or near the upper part of the windscreen.**
 - There shall be no metal objects (e.g. name badges, stickers, foil anti reflection (tinting) strips, sun visors, windshield wiper at rest) in front of, or close to the antenna, that can interfere with the communication.
 - The antenna shall be mounted so that its boresight approximately is parallel with the surface of the road.

DSC_20 The Antenna and The Communication shall operate within ERC 70-03, tested against the appropriate parameters of EN 300 674-1 as described in section 45. The Antenna and the Communication can implement mitigation techniques against the risk of wireless interference as described in ECC report 228 using e.g., filters in the CEN DSRC 5.8 GHz communication.

DSC_21 The DSRC antenna shall be connected to the DSRC-VU facility either directly within the module mounted to or close to the windshield, or through a dedicated cable constructed in a manner to make illegal disconnection difficult. Disconnection of or interference with the functioning of Antenna shall be a violation of ~~Regulation (EU) No. 165/2014~~ **this Agreement**. Deliberate masking or otherwise detrimentally affecting the operational performance of the Antenna shall be construed as a violation of ~~this Regulation (EU) No. 165/2014~~ **Agreement**.

DSC_22 The form factor of the antenna is not defined and shall be a commercial decision, so long as the fitted DSRC-VU meets the conformance requirements defined in section 5 below. The antenna shall be positioned as determined in DSC_19 and ~~shown in figure 14.4 (oval line) and~~ it efficiently **supports** the use cases described in ~~3.1.2 and 3.1.3~~ **4.1.2 and 4.1.3**.

Figure 14.4

Example of positioning of the 5.8 GHz DSRC antenna in the windshield of regulated vehicles

The form factor of *the REDCR* and its antenna may vary according to the circumstances of the reader (tripod mounted, hand held, vehicle mounted, etc.) and the *modus operandi* employed by the agent of the competent control authorities.

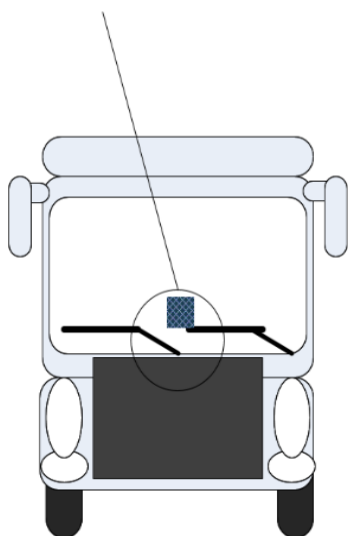
A display and/or notification function is used to present the results of the remote communication function to the agent of the competent control authorities. A display may be provided on a screen, as a printed output, an audio signal, or a combination of such notifications. The form of such display and/or notification is a matter of the requirements of the agents of the competent control authorities and equipment design and is not specified within this ~~Appendix~~ **sub-appendix**.

DSC_23 The design and form factor of the *REDCR* shall be a function of commercial design, operating within ERC 70-03, and the design and performance specifications defined in this ~~Appendix~~ **sub-appendix**, (section 5.3.2), thus providing the marketplace maximum flexibility to design and provide equipment to cover the specific interrogation scenarios of any particular competent control authority.

DSC_24 The design and form factor of the *DSRC-VU* and its positioning inside or outside the VU shall be a function of commercial design, operating within ERC 70-03 and the design and performance specifications defined in this ~~Appendix~~ **sub-appendix** (section 5.3.2) and within this Clause (5.1).

DSC_25 However, the *DSRC-VU* shall be reasonably capable to accept data concept values from other intelligent vehicle equipment by means of an open industry standard connection and protocols. (For example from weigh on board equipment), so long as such

CEN-DSRC Antenna Location



data concepts are identified by unique and known application identifiers/file names, and the instructions to operate such protocols shall be made available to the **laboratory competent for interoperability tests** ~~European Commission~~, and available without charge to manufacturers of relevant equipment.

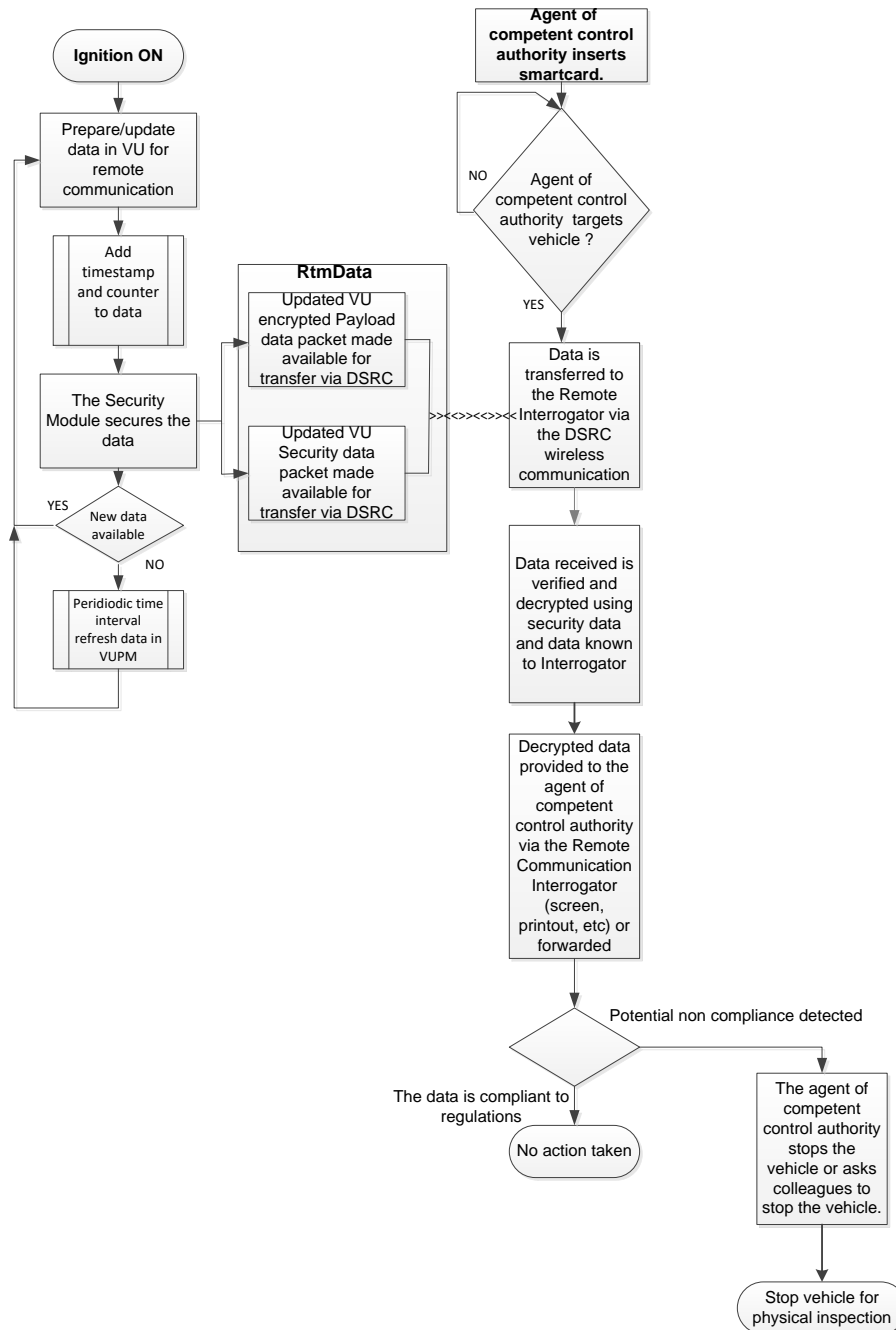
5.2. 4.2 Workflow

5.2.1 5.1 Operations

The workflow of operations is represented in Figure 14.5.

Figure 14.5

Workflow for remote communication function



The steps are described below:

- (a) Whenever the vehicle is in operation (ignition ON) the tachograph is providing data to the VU function. The VU function prepares *the Data* for the remote

communication function (encrypted) and updates the *VUPM* held in the memory of the *DSRC-VU* (as defined in 4.1.1.1 - 4.1.1.2). *The Data* collected shall be formatted as determined in 5.4.4 – 5.4.5 below.

(b) On every occasion that *the Data* is updated, the timestamp defined in the security data concept shall be updated.

(c) The *VUSM* function secures the data in accordance with the procedures determined in ~~Appendix~~ **Sub-appendix 11**.

(d) On every occasion that *the Data* is updated (see 4.1.1.1 - 4.1.1.2), *the Data* shall be transferred to the *DSRC-VU*, where it replaces any previous data, in order that updated current data (*the Data*) shall always be available to be provided in the event of an interrogation by an *REDCR*. When supplied by the *VU* to the *DSRC-VU* *the Data* shall be identifiable by the filename *RTMData* or by *ApplicationID* and *Attribute* identifiers.

(e) If an agent of the competent control authorities wishes to target a vehicle and collect *the Data* from the targeted vehicle, the agent of the competent control authorities shall first insert his/her smartcard in *the REDCR* to enable *the Communication* and to allow the *SM-REDCR* to verify its authenticity and decrypt the data.

(f) The agent of the competent control authority then targets a vehicle and requests the data through remote communication. *The REDCR* opens a 5.8 GHz *DSRC* interface session with the *DSRC-VU* of the targeted vehicle, and requests *the Data*. *The Data* is transferred to *the REDCR* through the wireless communication system as a *DSRC Attribute* using the *Application service GET* as defined in 5.4. The *Attribute* contains the encrypted payload data values and the *DSRC security data*.

(g) The data is analyzed by the *REDCR* equipment and provided to the agent of the competent control authority.

(h) The agent of the competent control authority uses the data to assist in a decision of whether or not to stop for a detailed inspection, or ask another agent of the competent control authority to stop the vehicle.

5.2.2 Interpretation of the Data received via the DSRC communication

DSC_26 Data received across the 5.8 GHz interface shall carry the meaning and import defined in 5.4.4 and 5.4.5 below and only that meaning and import, and shall be understood within the objectives defined therein. In accordance with the provision of ~~Regulation (EU) No. 165/2014~~ **prevailing legislation in each Contracting Party**, *the Data* shall be used only to provide relevant information to a competent control authority to assist them to determine which vehicle should be stopped for physical inspection, and shall be subsequently destroyed in accordance with ~~Article 9 of Regulation (EU) No. 165/2014~~ **the prevailing legislation in each Contracting Party**.

5.3. DSRC Physical interface parameters for remote communication

5.3.1 Location constraints

DSC_27 The remote interrogation of vehicles using a 5.8GHz *DSRC* interface should not be used within 200 metres of an operational 5.8 GHz *DSRC* gantry.

5.3.2 Downlink and uplink parameters

DSC_28 The equipment used for remote tachograph monitoring shall conform to and operate within ERC70-03 and the parameters defined in Tables 14.1 and 14.2 below.

DSC_29 Further, to ensure compatibility with the operational parameters of other standardised 5.8 GHz DSRC systems, the equipment used for remote tachograph monitoring shall conform to parameters from EN 12253 and EN 13372.

Namely:

Table 14.1

Downlink parameters

<i>Item No.</i>	<i>Parameter</i>	<i>Value(s)</i>	<i>Remark</i>
D1	Downlink Carrier Frequencies	There are four alternatives which may be used by an REDCR: 5.7975 GHz 5.8025 GHz 5.8075 GHz 5.8125 GHz	Within ERC 70-03. Carrier Frequencies may be selected by the implementer of the roadside system and need not be known in the DSRC-VU (Consistent with EN 12253, EN 13372)
D1a (*)	Tolerance of Carrier Frequencies	within ± 5 ppm	(Consistent with EN 12253)
D2(*)	RSU(REDCR) Transmitter Spectrum Mask	Within ERC 70-03. REDCR shall be according to Class B,C as defined in EN 12253 . No other specific requirement within this Annex Appendix	Parameter used for controlling interference between interrogators in proximity (as defined in EN 12253 and EN 13372).
D3	OBU(DSRC-VU) Minimum Frequency Range	5.795 – 5.815 GHz	(Consistent with EN 12253)
D4 (*) (**)	Maximum E.I.R.P.	Within ERC 70-03 (unlicensed) and within National Regulation Maximum +33 dBm	(Consistent with EN 12253)
D4a	Angular E.I.R.P. mask	According to declared and published specification of interrogator designer	(Consistent with EN 12253)
D5	Polarisation	Left hand circular	(Consistent with EN 12253)
D5a	Cross-Polarisation	XPD: In bore sight: (REDCR) RSU $t \geq 15$ dB (DSRC-VU) OBU $r \geq 10$ dB At -3 dB area: (REDCR) RSU $t \geq 10$ dB (DSRC-VU) OBU $r \geq 6$ dB	(Consistent with EN 12253)
D6 (*)	Modulation	Two level amplitude modulation.	(Consistent with EN 12253)

Item No.	Parameter	Value(s)	Remark
D6a (*)	Modulation Index	0.5 ... 0.9	(Consistent with EN 12253)
D6b	Eye Pattern	$\geq 90\%$ (time) / $\geq 85\%$ (amplitude)	
D7 (*)	Data Coding	FM0 "1" bit has transitions only at the beginning and end of the bit interval. "0" bit has an additional transition in the middle of the bit interval compared to the "1" bit.	(Consistent with EN 12253)
D8 (*)	Bit rate	500 kBit/s	(Consistent with EN 12253)
D8a	Tolerance of Bit Clock	better than ± 100 ppm	(Consistent with EN 12253)
D9(*)	Bit Error Rate (B.E.R.) for communication	$\leq 10^{-6}$ when incident power at OBU (DSRC-VU) is in the range given by [D11a to D11b].	(Consistent with EN 12253)
D10	Wake-up trigger for OBU (DSRC-VU)	OBU (DSRC-VU) shall wake up on receiving any frame with 11 or more octets (including preamble)	No special wake-up pattern is necessary. DSRC-VU may wake up on receiving a frame with less than 11 octets (Consistent with EN 12253)
D10a	Maximum Start Time	≤ 5 ms	(Consistent with EN 12253)
D11	Communication zone	Spatial region within which a B.E.R. according to D9a is achieved	(Consistent with EN 12253)
D11a (*)	Power Limit for communication (upper).	-24dBm	(Consistent with EN 12253)
D11b (*)	Power Limit for communication (lower).	Incident power: -43 dBm (boresight) -41 dBm (within -45° - $+45^\circ$ corresponding to the plane parallel to the road surface when the DSRC-VU later is installed in the vehicle (Azimuth))	(Consistent with EN 12253) Extended requirement for horizontal angles up to $\pm 45^\circ$, due to the use cases defined in this Annex Sub-appendix .

Item No.	Parameter	Value(s)	Remark
D12(*)	Cut-off power level of (DSRC-VU)	-60 dBm	(Consistent with EN 12253)
D13	Preamble	Preamble is mandatory.	(Consistent with EN 12253)
D13a	Preamble Length and Pattern	16 bits \pm 1 bit of FM0 coded "1" bits	(Consistent with EN 12253)
D13b	Preamble Wave form	An alternating sequence of low level and high level with pulse duration of 2 μ s. The tolerance is given by D8a	(Consistent with EN 12253)
D13c	Trailing Bits	The RSU (REDCR) is permitted to transmit a maximum of 8 bits after the end flag. An OBU (DSRC-VU) is not required to take these additional bits into account.	(Consistent with EN 12253)

(*) - Downlink parameters subject to conformance testing in accordance with relevant parameter test from EN 300 674-1

(**) – **The maximum value of the effective isotropically radiated power of the equipment used for remote monitoring of the tachograph must correspond to the requirements established on the territory of the Contracting Party.**

Table 14.2

Uplink parameters

Item No.	Parameter	Value(s)	Remark
U1 (*)	Sub-carrier Frequencies	A OBU (DSRC-VU) shall support 1.5 MHz and 2.0 MHz An RSU (REDCR) shall support 1.5 MHz or 2.0 MHz or both. U1-0: 1.5 MHz U1-1: 2.0 MHz	Selection of sub-carrier frequency (1.5 MHz or 2.0 MHz) depends on the EN 13372 profile selected.
U1a(*)	Tolerance of Sub- carrier Frequencies	within \pm 0.1%	(Consistent with EN 12253)
U1b	Use of Side Bands	Same data on both sides	(Consistent with EN 12253)
U2 (*)	OBU (DSRC-VU) Transmitter Spectrum Mask	According to EN12253 1) Out band power: see ETSI EN 300674-1 2) In band power: [U4a] dBm in 500 kHz 3) Emission in any other uplink channel: U2(3)-1 = -35 dBm in 500 kHz	(Consistent with EN 12253)

Item No.	Parameter	Value(s)	Remark
U4a (*)	Maximum Single Side Band E.I.R.P. (boresight)	Two options: U4a-0: -14 dBm U4a-1: -21 dBm	According to declared and published specification of equipment designer
U4b (*)	Maximum Single Side Band E.I.R.P. (35°)	Two options: - Not applicable - -17dBm	According to declared and published specification of equipment designer
U5	Polarisation	Left hand circular	(Consistent with EN 12253)
U5a	Cross Polarisation	XPD: In bore sight: (REDCR) RSU $r \geq 15$ dB (DSRC-VU) OBU $t \geq 10$ dB At -3 dB: (REDCR) RSU $r \geq 0$ dB (DSRC-VU) OBU $t \geq 6$ dB	(Consistent with EN 12253)
U6	Sub-Carrier Modulation	2-PSK Encoded data synchronised with sub-carrier: Transitions of encoded data coincide with transitions of sub-carrier.	(Consistent with EN 12253)
U6b	Duty Cycle	Duty Cycle: 50% $\pm \alpha$, $\alpha \leq 5\%$	(Consistent with EN 12253)
U6c	Modulation on Carrier	Multiplication of modulated sub-carrier with carrier.	(Consistent with EN 12253)
U7 (*)	Data Coding	NRZI (No transition at beginning of "1" bit, transition at beginning of "0" bit, no transition within bit)	(Consistent with EN 12253)
U8 (*)	Bit Rate	250 kbit/s	(Consistent with EN 12253)
U8a	Tolerance of Bit Clock	Within ± 1000 ppm	(Consistent with EN 12253)
U9	Bit Error Rate (B.E.R.) for communication	$\leq 10^{-6}$	(Consistent with EN 12253)
U11	Communication Zone	The spatial region within which the DSRC-VU is situated such that its transmissions are received by the REDCR with a B.E.R. of less than that given by U9a.	(Consistent with EN 12253)
U12a(*)	Conversion Gain (lower limit)	1 dB for each side band Range of angle: Circularly symmetric between bore sight and $\pm 35^\circ$ and within $-45^\circ - +45^\circ$ corresponding to the plane parallel to the road surface when the DSRC-VU later is installed in the vehicle (Azimuth)	Greater than the specified value range for horizontal angles up to $\pm 45^\circ$, due to the use cases defined in this Annex Appendix .
U12b(*)	Conversion Gain (upper limit)	10 dB for each side band	Less than the specified value range for each side band within a circular cone around boresight of $\square 45^\circ$ opening angle

Item No.	Parameter	Value(s)	Remark
U13	Preamble	Preamble is mandatory.	(Consistent with EN 12253)
U13a	Preamble Length and Pattern	32 to 36 μ s modulated with sub-carrier only, then 8 bits of NRZI coded "0" bits.	(Consistent with EN 12253)
U13b	Trailing Bits	The DSRC-VU is permitted to transmit a maximum of 8 bits after the end flag. A RSU (REDCR) is not required to take these additional bits into account.	(Consistent with EN 12253)

(*) - Uplink parameters subject to conformance testing in accordance with relevant parameter test from EN 300 674-1

5.3.3 ~~4.3.3~~ Antenna design

5.3.3.1 ~~4.3.3.1~~ REDCR antenna

DSC_30 The design of the REDCR antenna shall be a function of commercial design, operating within the limits defined in 5.3.2 which is adapted to optimise the reading performance of the DSRC-REDCR for the specific purpose and read circumstances in which the REDCR has been designed to operate.

5.3.3.2 ~~4.3.3.2~~ VU antenna

DSC_31 The design of the DSRC-VU antenna shall be a function of commercial design, operating within the limits defined in 5.3.2 which is adapted to optimise the reading performance of the DSRC-REDCR for the specific purpose and read circumstances in which the REDCR has been designed to operate.

DSC_32 The VU antenna shall be fixed to, or close to, the front windshield of the vehicle as specified in 5.1 above.

DSC_33 In the test environment in a workshop (see section 6.3), a DSRC-VU antenna, affixed according to 5.1 above, shall successfully connect with a standard test communication and successfully provide an RTM transaction as defined within this ~~Appendix~~ **sub-appendix**, at a distance between 2 and 10 meters, better than 99% of the time, averaged over 1000 read interrogations.

5.4. ~~4.4~~ DSRC Protocol requirements for RTM

5.4.1 ~~4.4.1~~ Overview

DSC_34 The transaction protocol to download *the Data* across the 5.8 GHz DSRC interface link shall be according to the following steps. This section describes a transaction flow under ideal conditions without retransmissions or communication interrupts.

NOTE: ~~The~~ the purpose of the initialisation phase (Step 1) is to set up the communication between the REDCR and DSRC-VUs that have entered the 5.8 GHz DSRC (master-slave) transaction zone but have not yet established communication with the REDCR, and to notify the application processes.

↳ **Step 1** Initialisation. The REDCR sends a frame containing a 'beacon service table' (BST) that includes the application identifiers (AIDs) in the service list that it supports. In the RTM application this will simply be the service with the AID value = 2 (Freight&Fleet). The DSRC-VU evaluates the received BST, and shall respond (see below) with the list of the

supported applications within the Freight&Fleet domain, or shall not respond if none are supported. If the *REDCR* does not offer AID=2, the *DSRC-VU* shall not answer to the *REDCR*.

↳ **Step 2** The *DSRC-VU* sends a frame containing a request for a private window allocation.

↳ **Step 3** The *REDCR* sends a frame containing a private window allocation.

↳ **Step 4** The *DSRC-VU* uses the allocated private window to send a frame containing its vehicle service table (VST). This VST includes a list of all the different application instantiations that this *DSRC-VU* supports in the framework of AID=2. The different instantiations shall be identified by means of uniquely generated EIDs, each associated with an Application Context Mark parameter value indicating the application and standard supported.

↳ **Step 5** Next the *REDCR* analyses the offered VST, and either terminates the connection (RELEASE) since it is not interested in anything the VST has to offer (i.e. it is receiving a VST from a *DSRC-VU* that is not supporting the RTM transaction), or, if it receives an appropriate VST it starts an app instantiation.

↳ **Step 6** To bring this about, the *REDCR* shall send a frame containing a command to retrieve the RTM data, identifying the RTM application instantiation by specifying the identifier corresponding to the RTM application instantiation (as specified by the *DSRC-VU* in the VST), and shall allocate a private window.

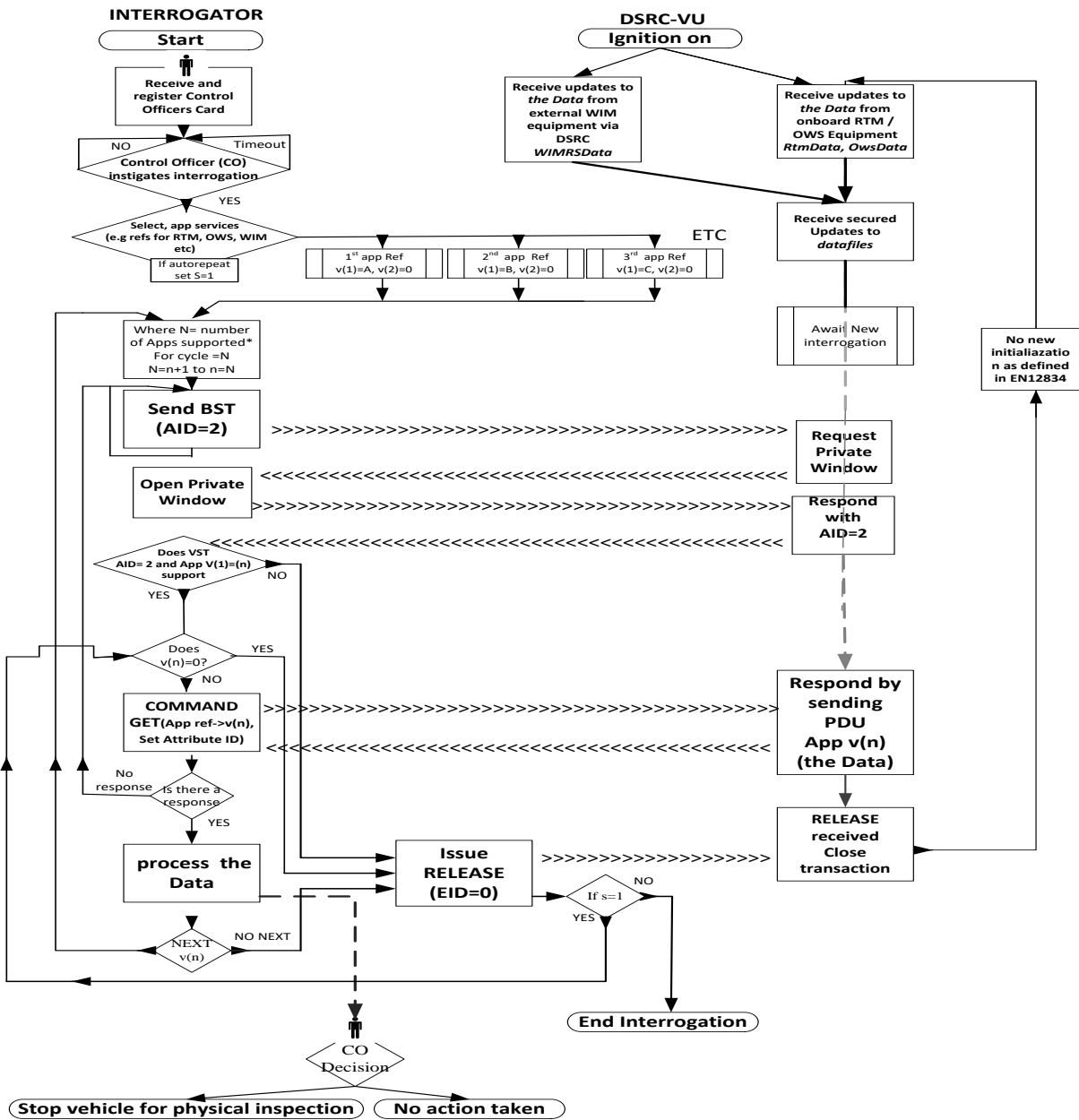
↳ **Step 7** The *DSRC-VU* uses the newly allocated private window to send a frame that contains the addressed identifier corresponding to the RTM application instantiation as provided in the VST, followed by the attribute *RtmData* (payload element + security element).

↳ **Step 8** If there are multiple services requested, the value 'n' is changed to the next service reference number and the process repeated.

↳ **Step 9** The *REDCR* confirms receipt of the data by sending a frame containing a RELEASE command to the *DSRC-VU* to terminate the session OR if it has failed to validate a successful receipt of the LDPU goes back to step 6.

See Figure 14.6 for a pictorial description of the transaction protocol.

Figure 14.6
RTM over 5.8 GHz DSRC process flow



5.4.2 4.4.2 Commands

DSC_35 The following commands are the only functions used in an RTM transaction phase

- **INITIALISATION.request:** A command, issued from the REDCR in the form of a broadcast with definition of applications that the REDCR supports.
- **INITIALISATION.response:** An answer from the DSRC-VU confirming the connection and containing a list of supported application instances with characteristics and information how to address them (EID).
- **GET.request:** A command, issued from *the REDCR* to the *DSRC-VU*, that specifies the application instantiation to be addressed by means of a defined EID, as received in the VST, instructing the *DSRC-VU* to send the selected attribute(s) with *the Data*. The objective of the GET command is for *the REDCR* to obtain *the Data* from the *DSRC-VU*.
- **GET.response:** An answer from the DSRC-VU that contains *the Data* requested.
- **ACTION.request ECHO:** A command, instructing the *DSRC-VU* to send back data from the *DSRC-VU* to *the REDCR*. The objective of the ECHO command is to enable workshops or type approval test facilities to test that the DSRC link is working without needing access to security credentials.
- **ACTION.response ECHO:** An answer from the DSRC VU on the ECHO command.
- **EVENT_REPORT.request RELEASE:** A command, instructing the *DSRC-VU* that the transaction is ended. The objective of the RELEASE command is to end the session with the *DSRC-VU*. On receipt of the RELEASE the *DSRC-VU* shall not respond to any further interrogations under the current connection. Note that according to EN 12834 a DSRC-VU will not connect twice to the same interrogator unless it has been out of the communication zone for 255 seconds or if the Beacon ID of the interrogator is changed.

5.4.3 4.4.3 Interrogation command sequence

DSC_36 From the perspective of the command and response sequence, the transaction is described as follows:

Sequence	Sender	Receiver	Description	Action
1	REDCR	> DSRC-VU	Initialisation of the communication link – Request	REDCR broadcasts BST
2	DSRC-VU	> REDCR	Initialisation of the communication link – Response	If BST supports AID=2 then DSRC-VU Requests a private window
3	REDCR	> DSRC-VU	Grants a private window	Sends Frame containing private window allocation
4	DSRC-VU	> REDCR	Sends VST	Sends Frame comprising VST
5	REDCR	> DSRC-VU	Sends GET.request for data in Attribute for specific EID	

Sequence	Sender	Receiver	Description	Action
6	DSRC-VU	> REDCR	Sends GET.response with requested Attribute for specific EID	Sends Attribute (RTMData, OWSData ...) with data for specific EID
7	REDCR	> DSRC-VU	Sends GET.request for data of other Attribute" (if appropriate)	
8	DSRC-VU	> REDCR	Sends GET.response with requested Attribute	Sends Attribute with data for specific EID
9	REDCR	> DSRC-VU	Acknowledges successful receipt of data	Sends RELEASE command which closes transaction
10	DSRC-VU		Closes transaction	

An example of the transaction sequence and contents of the exchanged frames is defined in clauses 5.4.7 and 5.4.8

5.4.4 4.4.4 Data structures

DSC_37 The semantic structure of the Data when passed across the 5.8 GHz DSRC interface shall be consistent with what described in this ~~Appendix~~ **sub-appendix**. The way these data are structured is specified in this clause.

DSC_38 The payload (RTM data) consists of the concatenation of

1. EncryptedTachographPayload data, which is the encryption of the TachographPayload defined in ASN.1 in section 5.4.5. The method of encryption is described in ~~Appendix~~ **Sub-appendix 11**
2. DSRCSecurityData, specified in ~~Appendix~~ **Sub-appendix 11**.

DSC_39 The RTM Data is being addressed as RTM Attribute=1 and is transferred in the RTM container =10.

DSC_40 The RTM Context Mark shall identify the supported standard part in the TARV series of standards (RTM corresponds to Part 9)

The ASN.1 module definition for the DSRC data within the RTM application is defined as follows:

```
TarvRtm {iso(1) standard(0) 15638 part9(9)
version1(1)} DEFINITIONS AUTOMATIC TAGS
 ::= BEGIN
IMPORTS
-- Imports data attributes and elements from EFC which are used for RTM
LPN
FROM EfcDsrcApplication {iso(1) standard(0) 14906 application(0) version5(5)}

-- Imports function parameters from the EFC Application Interface Definition
SetMMIRq
FROM EfcDsrcApplication {iso(1) standard(0) 14906 application(0) version5(5)}

-- Imports the L7 DSRCData module data from the EFC Application Interface Definition
Action-Request, Action-Response, ActionType, ApplicationList, AttributeIdList, AttributeList,
Attributes,
BeaconID, BST, Dsrc-EID, DSRCApplcationEntityID, Event-Report-Request, Event-Report-Response,
EventType, Get-Request, Get-Response, Initialisation-Request, Initialisation-Response,
ObeConfiguration, Profile, ReturnStatus, Time, T-APDUs, VST
FROM EfcDsrcGeneric {iso(1) standard(0) 14906 generic(1) version5(5)};
-- Definitions of the RTM functions:
RTM-InitialiseComm-Request ::= BST
RTM-InitialiseComm-Response ::= VST
RTM-DataRetrieval-Request ::= Get-Request (WITH COMPONENTS {fill (SIZE(1)), eid, accessCredentials
ABSENT, iid ABSENT, attrIdList})
RTM-DataRetrieval-Response ::= Get-Response {RtmContainer} (WITH COMPONENTS {..., eid, iid ABSENT})
```

```

RTM-TerminateComm ::= Event-Report-Request {RtmContainer} (WITH COMPONENTS {mode (FALSE), eid (0),
eventType (0)})
RTM-TestComm-Request ::= Action-Request {RtmContainer} (WITH COMPONENTS {..., eid (0), actionType
(15), accessCredentials ABSENT, iid ABSENT})
RTM-TestComm-Response ::= Action-Response {RtmContainer} (WITH COMPONENTS {..., fill (SIZE(1)), eid
(0), iid ABSENT})
-- Definitions of the RTM attributes:
RtmData ::= SEQUENCE {
  encryptedTachographPayload OCTET STRING (SIZE(67)) (CONSTRAINED BY { -- calculated encrypting
TachographPayload as per Appendix Sub-appendix 11 --}),
  DSRCSecurityData OCTET STRING
}
TachographPayload ::= SEQUENCE {
  tp15638VehicleRegistrationPlate LPN -- Vehicle Registration Plate using the data structure
from ISO 14906, but for the RTM application the LPN is fixed
to 17 bytes (no length determinant) as per EN 15509-17.
  tp15638SpeedingEvent BOOLEAN, -- 1= Irregularities in speed (see Annex
Appendix 1C)
  tp15638DrivingWithoutValidCard BOOLEAN, -- 1= Invalid card usage (see Annex Appendix
1C)
  tp15638DriverCard BOOLEAN, -- 0= Indicates a valid driver card (see Annex
Appendix 1C)
  tp15638CardInsertion BOOLEAN, -- 1= Card insertion while driving (see Annex
Appendix 1C)
  tp15638MotionDataError BOOLEAN, -- 1= Motion data error (see Annex Appendix 1C)
  tp15638VehicleMotionConflict BOOLEAN, -- 1= Motion conflict (see Annex Appendix 1C)
  tp156382ndDriverCard BOOLEAN, -- 1= Second driver card inserted (see Annex
Appendix 1C)
  tp15638CurrentActivityDriving BOOLEAN, -- 1= other activity selected;
-- 0= driving selected
  tp15638LastSessionClosed BOOLEAN, -- 1= improperly, 0= properly, closed
  tp15638PowerSupplyInterruption INTEGER (0..127), -- Supply interrupts in the last 10 days
  tp15638SensorFault INTEGER (0..255), -- eventFaultType as per data dictionary
-- All subsequent time related types as defined in Annex Appendix 1C.
  tp15638TimeAdjustment INTEGER (0..4294967295), -- Time of the last time adjustment
  tp15638LatestBreachAttempt INTEGER (0..4294967295), -- Time of last breach attempt
  tp15638LastCalibrationData INTEGER (0..4294967295), -- Time of last calibration data
  tp15638PrevCalibrationData INTEGER (0..4294967295), -- Time of previous calibration
data
  tp15638DateTachoConnected INTEGER (0..4294967295), -- Date tachograph connected

  tp15638CurrentSpeed INTEGER (0..255), -- Last current recorded speed
  tp15638Timestamp INTEGER (0..4294967295) -- Timestamp of current record2
  tp15638LatestAuthenticatedPosition tp15638LatestAuthenticatedPosition INTEGER (0..4294967295), -- Time of latest authenticated
position
  tp15638ContinuousDrivingTime INTEGER (0..255), -- Continuous driving time of the driver
  tp15638DailyDrivingTimeShift INTEGER (0..255), -- Longest daily driving time of the
driver for the current ongoing and previous RTM-shift
  tp15638DailyDrivingTimeWeek INTEGER (0..255), -- Longest daily driving time of the
driver within the current ongoing week
  tp15638WeeklyDrivingTime INTEGER (0..255), -- Weekly driving time of the driver
  tp15638FortnightlyDrivingTime INTEGER (0..255) -- Fortnightly driving time of the driver
}
Rtm-ContextMark ::= SEQUENCE {
  standardIdentifier StandardIdentifier, -- identifier of the TARV part and its version
  RtmCommProfile INTEGER {
    C1 (1),
    C2 (2)
  } (0..255) DEFAULT 1
}
RtmTransferAck ::= INTEGER {
  Ok (1),
  NoK (2)
} SIZE(1..255)
StandardIdentifier ::= OBJECT IDENTIFIER
RtmContainer ::= CHOICE {
  integer [0] INTEGER,
  bitstring [1] BIT STRING,
  octetstring [2] OCTET STRING (SIZE (0..127, ...)),
  universalString [3] UniversalString,
  beaconId [4] BeaconID,
  t-apdu [5] T-APDUs,

```

¹⁷ if a LPN contains an AlphabetIndicator LatinAlphabetNo2 or latinCyrillicAlphabet, the special characters are remapped at the road interrogator unit applying special rules according to Appendix E of ISO/DIS 14906.2

```

dsrcApplicationEntityId [6] DsrcApplicationEntityID,
dsrcAseId [7] Dsrc-EID,
attrIdList [8] AttributeIdList,
attrList [9] AttributeList{RtmContainer},
rtmData [10] RtmData,
rtmContextmark [11] Rtm-ContextMark,
reserved12 [12] NULL,
reserved13 [13] NULL,
reserved14 [14] NULL,
time [15] Time,
-- values from 16 to 255 reserved for ISO/CEN usage
}}

```

END

5.4.5 4.4.104 Elements of RtmData RTM data, actions performed and definitions

DSC_41 The data values to be calculated by the VU and used to update the secured data in the DSRC-VU shall be calculated according to the rules defined in Table 14.3:

Table 14.3

Elements of RtmData, actions performed and definitions

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM1 Vehicle Registration Plate	The VU shall set the value of the <i>tp15638VehicleRegistrationPlate</i> data element RTM1 from the recorded value of the data type <i>VehicleRegistrationIdentification</i> as defined in Appendix Sub-appendix 1 <i>VehicleRegistrationIdentification</i>	Vehicle Registration Plate expressed as a string of characters tp15638VehicleRegstrati onPlate LPN, --Vehicle Registration Plate imported from ISO 14906 with the limitation specified in EN 15509 which is a SEQUENCE comprising Country Code followed by an alphabet indicator followed by the plate number itself, which is always 14 octets (padded with zero's) so the EN 15509 LPN type length is always 17 octets, of which 14 are the "real" plate number.
RTM2 Speeding Event	The VU shall generate a boolean value for data element RTM2 <i>tp15638SpeedingEvent</i> . The <i>tp15638SpeedingEvent</i> value shall be calculated by the VU from the number of Over Speeding Events recorded in the VU in the last 10 days of occurrence, as defined in Annex Appendix 1C . If there is at least one <i>tp15638SpeedingEvent</i> in the last 10 days of occurrence, the <i>tp15638SpeedingEvent</i> value shall be set to TRUE. ELSE if there are no events in the last 10 days of occurrence, the <i>tp15638SpeedingEvent</i> shall be set to FALSE.	1 (TRUE) - if the most recent over speeding event ended within the last 10 days or is still ongoing; Indicates irregularities in speed within last 10 days of occurrence 0 (FALSE): in any other case. tp15638speedingEvent BOOLEAN,

(1) RTM Data Element	(2) Action performed by the VU		(3) ASN.1 definition of data
RTM3 Driving Without Valid Card	<p>The VU shall generate a boolean value for data element RTM3 tp15638DrivingWithoutValidCard.</p> <p>The VU shall assign a value of True to the tp15638DrivingWithoutValidCard variable if the VU data has recorded at least one event in the last 10 days of occurrence of type “Driving without an appropriate card” an appropriate card²² event has been recorded in the VU within the last 10 days as defined in Annex Appendix 1C. Annex Appendix 1C.</p> <p>ELSE if there are no events in the last 10 days of occurrence, the tp15638DrivingWithoutValidCard variable shall be set to FALSE.</p>	<p>1 (TRUE): if the most recent driving without an appropriate card event ended within the last 10 days or is still ongoing;</p> <p>0 (FALSE): in any other case. Indicates invalid card usage</p>	tp15638DrivingWithoutValidCard BOOLEAN,
RTM4 Valid Driver Card	<p>The VU shall generate a boolean value for data element RTM4 tp15638DriverCard on the basis of the inserted valid driver card in the driver slot data stored in the VU and defined in Appendix Sub-appendix 1.</p> <p>If no valid driver card is present the VU shall set the variable to TRUE</p> <p>ELSE if a valid driver card is present the VU shall set the variable to FALSE</p>	<p>1 (TRUE): if no valid driver card is present in the driver slot of the VU;</p> <p>0 (FALSE): =Indicates if a valid driver card is present in the driver slot of the VU. Indicates if a valid driver card is present in the driver slot of the VU.</p>	tp15638DriverCard BOOLEAN,
RTM5 Card Insertion while Driving	<p>The VU shall generate a boolean value for data element RTM5 tp15638CardInsertion..</p> <p>The VU shall assign a value of TRUE to the tp15638CardInsertion variable if the VU data has recorded in the last 10 days of occurrence at least one event of type “Card insertion while driving event has been recorded in the VU within the last 10 days,²² as defined in Annex Appendix 1C. Annex Appendix 1C.</p> <p>ELSE if there are no such events in the last 10 days of occurrence, the tp15638CardInsertion variable shall be set to FALSE.</p>	<p>1 (TRUE) = if the most recent Indicates card insertion while driving event has occurred within last 10 days; Indicates card insertion while driving event has occurred within last 10 days;</p> <p>0 (FALSE): in any other case. of occurrence</p>	tp15638CardInsertion BOOLEAN,
RTM6 Motion Data Error	<p>The VU shall generate a boolean value for data element RTM6.</p> <p>The VU shall assign a value of TRUE to the tp15638MotionDataError variable if</p>	<p>1 (TRUE): if the most recent =Indicates motion data error event ended within last 10 days Indicates motion data error event ended within last 10 days or is still ongoing;</p>	tp15638MotionDataError BOOLEAN,

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM7 Vehicle Motion Conflict	<p>the VU data has in the last 10 days of occurrence recorded at least one event of type "Motion data error event" has been recorded in the VU within the last 10 days as defined in Annex Appendix 1C.</p> <p>ELSE if there are no such events in the last 10 days of occurrence, the tp15638MotionDataError variable shall be set to FALSE.</p> <p>The VU shall generate a boolean value for data element RTM7.</p> <p>The VU shall assign a value of TRUE to the tp15638VehicleMotionConflict variable if the VU data has in the last 10 days recorded at least one event of type "Vehicle Motion Conflict event has been recorded in the VU within the last 10 days (value '0A'H).</p> <p>ELSE if there are no events in the last 10 days of occurrence, the tp15638vehicleMotionConflict variable shall be set to FALSE.</p>	<p>0 (FALSE): in any other case of occurrence</p> <p>1 (TRUE): if the most recent vehicle motion conflict event ended within last 10 days or is still ongoing</p> <p>0 (FALSE): in any other case of occurrence</p> <p>tp15638VvehicleMotionConflict BOOLEAN,</p>
RTM8 2nd Driver Card	<p>The VU shall generate a boolean value for data element RTM8 on the basis of Annex Appendix 1C ("Driver Activity Data" CREW and CO-DRIVER).</p> <p>If a 2nd valid co-driver card is present the VU shall set the value of RTM8 variable to TRUE</p> <p>ELSE if a 2nd valid driver card is not present the VU shall set the variable to FALSE</p>	<p>1 (TRUE): if= Indicates a valid second co-driver card is present in the VU</p> <p>2 (FALSE): if no valid co-driver card is present in the VU. inserted</p> <p>tp156382ndDriverCard BOOLEAN,</p>
RTM9 Current Activity	<p>The VU shall generate a boolean value for data element RTM9.</p> <p>If the current activity is recorded in the VU as any activity other than "DRIVING" as defined in Annex Appendix 1C the VU shall set the value of RTM9 variable to TRUE</p> <p>ELSE if the current activity is recorded in the VU as "DRIVING" the VU shall set the variable to FALSE</p>	<p>1 (TRUE): = other activity selected;</p> <p>0 (FALSE): =driving selected</p> <p>tp15638eCurrentActivityDriving BOOLEAN</p>

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM10 Last Session Closed	<p>The VU shall generate a boolean value for data element RTM10.</p> <p>If the last card session was not properly closed as defined in Annex-Appendix 1C the VU shall set the value of RTM10 variable to TRUE.</p> <p>ELSE if the last card session was properly closed the VU shall set the variable to FALSE</p>	<p>1 (TRUE): at least one of the inserted cards has triggered a last card session not correctly closed event; = improperly closed 0 (FALSE): None of the inserted cards has triggered a last card session not correctly closed event. = properly closed</p>
RTM11 Power Supply Interruption	<p>The VU shall generate an integer value for data element RTM11.</p> <p>The VU shall assign a value for the tp15638PowerSupplyInterruption variable equal to the number of the recorded longest power supply interruption events stored in the VU within the last 10 days as defined in Appendix 1C.² according to Article 9, Reg (EU) 165/2014 of type "Power supply interruption" as defined in Annex 1C.</p> <p>If no power supply interruption event has been recorded in the VU within the last 10 days, it shall set the value of RTM11 to 0.</p> <p>ELSE if in the last 10 days of occurrence there are have been no Power supply interruption events the value of the integer shall be set to 0.</p>	<p>-- Number of the recorded power supply Interruption events within the last 10 days of occurrence</p>
RTM12 Sensor Fault	<p>The VU shall generate an integer value for data element RTM12.</p> <p>The VU shall assign to the variable sensorFault a value of:</p> <ul style="list-style-type: none"> - 1 if an event of type '35H' Sensor fault ended duringhas been recorded in the last 10 days or is still ongoing, - 2 if an event of type GNSS receiver fault (either internal or external with enum values 51'H'36'H or 52'H'37' H) ended duringhas been recorded in the last 10 days or is still ongoing. - 3 if an event of type 53'H'External'0E'H Communication error with the external GNSS communication fault facility event ended duringhas been recorded in the last 10 days or is still ongoingof occurrence. 	<p>--sensor fault one octet as per data dictionary</p>

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM13 Time Adjustment	<p>-4 If both Sensor Fault and GNSS receiver faults ended during have been recorded in the last 10 days or are still ongoing of occurrence.</p> <p>-5 If both Sensor Fault and Communication error with the external GNSS facility event ended during have been recorded in the last 10 days ir are still ongoing of occurrence.</p> <p>-6 If both GNSS receiver fault and Communication error with the external GNSS facility event ended during have been recorded in the last 10 days or are still ongoing of occurrence.</p> <p>-7 If all three sensor faults ended during have been recorded in the last 10 days or are still ongoing of occurrence.</p> <p>ELSE it shall assign a value of 0 if no events have ended during been recorded in the last 10 days or is still ongoing, the VU shall set the value of RTM12 to 0. of occurrence</p> <p>The VU shall generate an integer value (timeReal from Appendix Sub- appendix 1) for data element RTM13 on the basis of the presence of Time Adjustment data as defined in Annex Appendix 1C.</p> <p>The VU shall assign set the value of RTM13 to the time at which the last time adjustment data event has occurred.</p> <p>ELSE if no “Time Adjustment” event as defined in Annex Appendix 1C is present in the VU data, it shall set the the value of RTM13 to 0.</p>	<p>oldTimeValue of the most recent last time adjustment</p> <p>tp15638TimeAdjustment INTEGER(0..4294967295),</p>
RTM14 Security Breach Attempt	<p>The VU shall generate an integer value (timeReal from Appendix Sub- appendix 1) for data element RTM14 on the basis of the presence of a S security breach attempt event as defined in Annex Appendix 1C.</p> <p>The VU shall set the value of the time of the latest security breach attempt event recorded by the VU.</p>	<p>Beginning Ttime of the latest stored security breach attempt event</p> <p>tp15638LatestBreachAttempt INTEGER(0..4294967295),</p> <p>—Default value =0x00FF</p>

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM15 Last Calibration	<p>ELSE if no “security breach attempt” event as defined in Appendix 1C is present in the VU data it shall set the value of RTM14 to 0x00FF.</p> <p>The VU shall generate an integer value (timeReal from Appendix Sub-appendix 1) for data element RTM15 on the basis of the presence of Last Calibration data as defined in Annex Appendix 1C and Sub-appendix 1. The VU shall set the value of time of the latest two calibrations (RTM15 to the oldTimeValue of the latest calibration record and RTM16), which are set in VuCalibrationData defined in Sub Appendix appendix 1. oldTimeValue of the most recent calibration record data If there has been no calibration, the VU shall set the value of RTM15 to 0 the timeReal of the latest calibration record.</p>	tp15638LastCalibrationData INTEGER(0..4294967295),
RTM16 Previous Calibration	<p>The VU shall generate an integer value (timeReal from Appendix Sub-appendix 1) for data element RTM16 on the basis of the calibration record preceding that of the last calibration oldTimeValue of the previous calibration record preceding the most recent calibration record data</p> <p>ELSE if there has been no previous calibration – The VU shall set the value of RTM16 to the oldTimeValue of the calibration record preceding the last calibration. If there has been no previous calibration, the VU shall set the value of RTM16 to 0.</p>	tp15638PrevCalibrationData INTEGER(0..4294967295),
RTM17 Date Tachograph Connected	<p>For data element RTM17 – The VU shall generate an integer value (timeReal from Appendix Sub-appendix 1) for data element RTM17. Date of the first calibration of the VU in the current vehicle tachograph connected</p> <p>The VU shall set the value of RTM17 to the date – the time of first calibration – the initial installation of the VU in the current vehicle. to the date – the time of first calibration – the initial installation of the VU in the current vehicle.</p> <p>The VU shall extract this data from the VuCalibrationData (Appendix Sub-appendix 1) from the vuCalibrationRecords with CalibrationPurpose equal to: '03'H</p>	tp15638DateTachoConnected INTEGER(0..4294967295),

(1) RTM Data Element	(2) Action performed by the VU		(3) ASN.1 definition of data
RTM18 Current Speed	<p>If there has been no previous calibration, the VU shall set the value of RTM17 to 0.</p> <p>The VU shall generate an integer value for data element RTM18.</p>	Last current recorded speed	tp15638CurrentSpeed INTEGER (0..255),
RTM19 Timestamp	<p>The VU shall set the value of of RTM18 to the last current recorded speed at the time of the latest update of the RtmData.</p> <p>For data element RTM19The VU shall generate an integer value for data element RTM19 (timeReal from Appendix Sub-appendix 1).</p> <p>The VU shall set the value of of RTM19 to the time of the latest update of the RtmData.</p>	Timestamp of current TachographPayload record	tp15638Timestamp INTEGER(0..4294967295),
RTM20 Time at which the latest authenticated vehicle position was available	<p>The VU shall generate an integer value (timeReal from Sub-appendix 1) for data element RTM20.</p> <p>The VU shall set the value of RTM20 to the time at which the latest authenticated vehicle position was available from the GNSS receiver.</p> <p>If no authenticated vehicle position was available ever from the GNSS receiver the VU shall set the value of RTM20 to 0.</p>	Timestamp of the latest authenticated vehicle position	tp15638LatestAuthenticatedPosition INTEGER(0..4294967295),
RTM21 Continuous driving time	<p>The VU shall generate an integer value for data element RTM21.</p> <p>The VU shall set the value of RTM21 to the ongoing continuous driving time of the driver.</p>	Continuous driving time of the driver, encoded as an integer value.	tp15638ContinuousDrivingTime INTEGER (0..255),
		<p>Length: 1 byte</p> <p>Resolution: 2 minutes/bit</p> <p>No offset</p> <p>Data range: 0 to 250</p> <p>A value of 250 shall indicate that the continuous driving time of the driver is equal or greater than 500 minutes.</p> <p>Values 251 to 254 are not used.</p> <p>Value 255 indicates that the information is not available.</p>	

(1) RTM Data Element	(2) Action performed by the VU		(3) ASN.1 definition of data
RTM22 Longest daily driving time for the ongoing and previous RTM-shift, calculated in accordance with the Annex to Sub-appendix 14	The VU shall generate an integer value for data element RTM22. The VU shall set the value of RTM22 to the longer of the two daily driving times of the driver, being either the ongoing or the previous RTM-shift.	Daily driving time of the driver, encoded as an integer value. Length: 1 byte Resolution: 4 minutes/bit No offset Data range: 0 to 250 A value of 250 shall indicate that the daily driving time of the driver is equal or greater than 1000 minutes. Values 251 to 254 are not used. Value 255 indicates that the information is not available.	tp15638DailyDrivingTimeShift INTEGER (0..255),
RTM23 Longest daily driving time within the ongoing week, calculated in accordance with the Annex to Sub-appendix 14	The VU shall generate an integer value for data element RTM23. The VU shall set the value of RTM23 to the longest daily driving time of the driver, being either the ongoing RTM-shift or any completed RTM-shift having started or finished in the ongoing week.	Daily driving time of the driver, encoded as an integer value. Length: 1 byte Resolution: 4 minutes/bit No offset Data range: 0 to 250 A value of 250 shall indicate that the daily driving time of the driver is equal or greater than 1000 minutes. Values 251 to 254 are not used. Value 255 indicates that the information is not available.	tp15638DailyDrivingTimeWeek INTEGER (0..255),
RTM24 Weekly driving time, calculated in accordance with the Annex to Sub-appendix 14	The VU shall generate an integer value for data element RTM24. The VU shall set the value of RTM24 to the weekly driving time of the driver.	Weekly driving time of the driver, encoded as an integer value. Length: 1 byte Resolution: 20 minutes/bit No offset Data range: 0 to 250 A value of 250 shall indicate that the weekly driving time of the driver is equal or greater than 5000 minutes. Values 251 to 254 are not used. Value 255 indicates that the information is not available.	tp15638WeeklyDrivingTime INTEGER (0..255),

(1) RTM Data Element	(2) Action performed by the VU	(3) ASN.1 definition of data
RTM25 Fortnightly driving time, calculated in accordance with the Annex to Sub-appendix 14	The VU shall generate an integer value for data element RTM25. The VU shall set the value of RTM25 to the fortnightly driving time of the driver.	Fortnightly driving time of the driver, encoded as an integer value. Length: 1 byte Resolution: 30 minutes/bit No offset Data range: 0 to 250 A value of 250 shall indicate that the fortnightly driving time of the driver is equal or greater than 7500 minutes. Values 251 to 254 are not used. Value 255 indicates that the information is not available.

Note: RTM22, RTM23, RTM24 and RTM25 shall be computed according to the Annex to this sub-appendix

5.4.6 4.4.102 Data transfer mechanism

DSC_42 Payload data defined previously are requested by the REDCR after initialisation phase, and consequently transmitted by the *DSRC-VU* in the allocated window. The command GET is used by the REDCR to retrieve data.

DSC_43 For all DSRC exchanges, data shall be encoded using PER (Packed Encoding Rules) UNALIGNED, apart from **TachographPayload** and **OwsPayload**, which shall be encoded using OER (Octet Encoding Rules) defined in ISO/IEC 8825-7, Rec. ITU-T X.696.

5.4.7 4.4.103 Detailed DSRC transaction description

DSC_44 Initialisation is performed according to DSC_44 – DSC_48 and Tables 14.4 – 14.9. In the initialisation phase, the REDCR starts sending a frame containing a BST (Beacon Service Table) according to EN 12834 and EN 13372, 6.2, 6.3, 6.4 and 7.1 with settings as specified in the following Table 14.4.

Table 14.4 — Initialisation - BST frame settings

Field	Settings
Link Identifier	Broadcast address
BeaconId	As per EN 12834
Time	As per EN 12834
Profile	No extension, 0 or 1 to be used
MandApplications	No extension, EID not present, Parameter not present, AID= 2 Freight&Fleet
NonMandApplications	Not present
ProfileList	No extension, number of profiles in list = 0

Field	Settings
Fragmentation header	No fragmentation
Layer 2 settings	Command PDU, UI command

A practical example of the settings specified in Table 14.4, with an indication of bit encodings, is given in the following Table 14.5.

Table 14.5

Initialisation - BST frame contents example

Octet #	Attribute /Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Broadcast ID	1111 1111	Broadcast address
3	MAC Control Field	1010 0000	Command PDU
4	LLC Control field	0000 0011	UI command
5	Fragmentation header	1xxx x001	No fragmentation
6	BST	1000	Initialisation request
	SEQUENCE { OPTION indicator BeaconID SEQUENCE { ManufacturerId INTEGER (0..65535)}	0	NonMand applications not present
		xxx	Manufacturer Identifier
7		xxxx xxxx	
8		xxxx x	
	IndividualID INTEGER (0..134217727)	xxx	27 bit ID available for manufacturer
9		xxxx xxxx	
10		xxxx xxxx	
11		xxxx xxxx	
12	Time INTEGER (0..4294967295)	xxxx xxxx	32 bit UNIX real time
13		xxxx xxxx	
14		xxxx xxxx	
15		xxxx xxxx	
16	Profile INTEGER (0..127,...)	0000 0000	No extension. Example profile 0
17	MandApplications SEQUENCE (SIZE(0..127,...)) OF {	0000 0001	No extension, Number of mandApplications = 1
18	SEQUENCE { OPTION indicator OPTION indicator AID DSRCApplicationEntityID } }	0	EID not present
		0	Parameter not present
		00 0010	No extension. AID= 2
19	ProfileList SEQUENCE (0..127,...) OF Profile }	0000 0000	No extension, number of profiles in
20	FCS	xxxx xxxx	Frame check sequence
21		xxxx xxxx	
22	Flag	0111 1110	End Flag

DSC_45 A DSRC-VU, when receiving a BST, requires the allocation of a private window, as specified by EN 12795 and EN 13372, 7.1.1, with no specific RTM settings. Table 14.6 provides an example of bit encoding.

Table 14.6

Initialisation - Private window allocation request frame contents

Octet #	Attribute/Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	0110 0000	Private window request
7	FCS	xxxx xxxx	Frame check sequence
8		xxxx xxxx	
9	Flag	0111 1110	End Flag

DSC_46 The REDCR then answers by allocating a private window, as specified by EN 12795 and EN 13372, 7.1.1 with no specific RTM settings.

Table 14.7 provides an example of bit encoding.

Table 14.7

Initialisation - Private window allocation frame contents

Octet #	Attribute/Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	0010 s000	Private window allocation
7	FCS	xxxx xxxx	Frame check sequence
8		xxxx xxxx	
9	Flag	0111 1110	End Flag

DSC_47 The DSRC-VU, when receiving the private window allocation, sends its VST (Vehicle Service Table) as defined in EN 12834 and EN 13372, 6.2, 6.3, 6.4 and 7.1 with settings as specified Table 14.8, using the allocated transmission window.

Table 14.8

Initialisation - VST frame settings

Field	Settings
Private LID	As per EN 12834

<i>Field</i>	<i>Settings</i>
VST parameters	Fill=0, then for each supported application: EID present, parameter present, AID=2, EID as generated by the OBU
Parameter	No extension, Contains the RTM Context Mark
ObeConfiguration	The optional ObeStatus field may be present, but shall not be used by the RFDOR
Fragmentation	No fragmentation
Layer 2 settings	Command PDU, UI command

DSC_48 The DSRC-VU shall support the “Freight and Fleet” application, identified by the Application Identifier ‘2’. Other Application Identifiers may be supported, but shall not be present in this VST, as the BST only requires AID=2. The “Applications” field contains a list of the supported application instances in the DSRC-VU. For each supported application instantiation, a reference to the appropriate standard is given, made of an Rtm Context mark, which is composed of an OBJECT IDENTIFIER representing the related standard, its part (9 for RTM) and possibly its version, plus an EID that is generated by the DSRC-VU, and associated to that application instance.

A practical example of the settings specified in Table 14.8, with an indication of bit encodings, is given in Table 14.9.

Table 14.9

Initialisation – VST frame contents example

Octet #	Attribute/Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1100 0000	Command PDU
7	LLC Control field	0000 0011	UI command
8	Fragmentation header	1xxx x001	No fragmentation
9	VST SEQUENCE { Fill BIT STRING (SIZE(4))	1001	Initialisation response
		0000	Unused and set to 0
10	Profile INTEGER (0..127,...)	0000 0000	No extension. Example profile 0
11		0000 0001	No extension, 1 application
12	SEQUENCE { OPTION indicator OPTION indicator AID DSRCApplicationEntityID	1	EID present
		1	Parameter present
		00 0010	No extension. AID= 2 Freight&Fleet
13	EID Dsrc-EID	xxxx xxxx	Defined within the OBU and identifying the application instance.

14	Parameter Container {	0000 0010	No extension, Container Choice = 02, Octet string
15		0000 0110	No extension, Rtm Context Mark length = 6
16	Rtm-ContextMark ::= SEQUENCE standardIdentifier	0000 0101	First octet is 05H, which is its length. Subsequent 5 octets encode the Object Identifier of the supported standard, part and version. {ISO (1) Standard (0) TARV (15638) part9(9) Version2 (2)}
17		0010 1000	
18		1111 1010	
19		0001 0110	
20		0000 1001	
21		0000 0010	
22	ObeConfiguration Sequence { OPTION indicator	0	ObeStatus not present
23	EquipmentClass INTEGER (0..32767)	xxx xxxx xxxx xxxx	This field shall be used to carry manufacturer's indications about the software/hardware version of the DSRC interface
24	ManufacturerId INTEGER (0..65535)	xxxx xxxx xxxx xxxx	
25			
26	FCS	xxxx xxxx	Frame check sequence
27		xxxx xxxx	
28	Flag	0111 1110	End Flag

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1100 0000	
7	LLC Control field	0000 0011	UI command
8	Fragmentation header	1xxx x001	No fragmentation
9	VST SEQUENCE {	1001	Initialisation response
	Fill BIT STRING (SIZE(4))	0000	Unused and set to 0

Octet#	Attribute/Field	Bits in octet	Description
10	Profile INTEGER (0..127,...)	0000-0000	No extension. Example profile 0
11		0000-0001	No extension, 1 application
12	SEQUENCE { OPTION indicator OPTION indicator AID-DSRCApplicationEntityID	+ -+	EID present Parameter present
		-00 0010	No extension. AID= 2 Freight&Fleet
13	EID-Dsrc-EID	xxxx-xxxx	Defined within the OBU and identifying the application instance.
14	Parameter Container {	0000-0010	No extension, Container Choice = 02, Octet string
15		0000-1000	No extension, Rtm Context Mark length = 8
16	Rtm-ContextMark ::= SEQUENCE { StandardIdentifier	0000-0110	Object Identifier of the supported standard, part, and version. Example: ISO (1) Standard (0) TARV (15638) part0 (9) Version1 (1).
17	standardIdentifier	0000-0110	First octet is 06H, which is the Object Identifier. Second octet is 06H, which is its length. Subsequent 6 octets encode the example Object Identifier
18		0010-1000	Note that only one element of the sequence is present (the optional RtmCommProfile element is omitted)
19		1000-0000	
20		1111-1010	
21		0001-0110	
22		0000-1001	
23		0000-0001	
24	ObeConfiguration-Sequence { OPTION indicator	0	ObeStatus not present
	EquipmentClass INTEGER (0..32767)	-xxx-xxxx	
25		xxxx-xxxx	
26	ManufacturerId INTEGER (0..65535)	xxxx-xxxx	Manufacturer identifier for the DSRC-VU as described in ISO 14816 Register
27		xxxx-xxxx	
28	FCS	xxxx-xxxx	Frame check sequence
29		xxxx-xxxx	
30	Flag	0111-1110	End Flag

DCS_49 The REDCR then reads the data by issuing a GET command, conforming to the GET command defined in EN 13372, 6.2, 6.3, 6.4 and EN 12834, with settings as specified in Table 14.10.

Table 14.10
Presentation – GET request frame settings

<i>Field</i>	<i>Settings</i>
Invoker Identifier (IID)	Not present
Link Identifier (LID)	Link address of the specific DSRC-VU
Chaining	No
Element Identifier (EID)	As specified in the VST. No extension
Access Credentials	No
AttributeIdList	No extension, 1 attribute, AttributeID = 1 (RtmData)
Fragmentation	No
Layer2 settings	Command PDU, Polled ACn command

Table 14.11 shows an example of reading the RTM data.

Table 14.11
Presentation – Get Request frame example

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1010 s000	Command PDU
7	LLC Control field	n111 0111	Polled ACn command, n bit
8	Fragmentation header	1xxx x001	No fragmentation
9	Get.request SEQUENCE { OPTION indicator OPTION indicator OPTION indicator Fill BIT STRING(SIZE(1))	0110 0 0 1 0	Get request Access Credentials not present IID not present AttributeIdList present Set to 0.
10	EID INTEGER(0..127,...)	xxxx xxxx	The EID of the RTM application instance, as specified in the VST. No extension
11	AttributeIdList SEQUENCE OF	0000 0001	No extension, number of attributes = 1
12		0000 0001	AttributeId=1, RtmData. No extension
13	FCS	xxxx xxxx	Frame check sequence

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
14		xxxx xxxx	
15	Flag	0111 1110	End Flag

DSC_50 The DSRC-VU, when receiving the GET request, sends a GET response with the requested data conforming to the GET response defined in EN 13372, 6.2, 6.3, 6.4 and EN 12834, with settings as specified in Table 14.12.

Table 14.12

Presentation – GET response frame settings

<i>Field</i>	<i>Settings</i>
Invoker Identifier	Not present
Link Identifier (LID)	As per EN 12834
Chaining	No
Element Identifier	As specified in the VST.
Access Credentials	No
Fragmentation	No
Layer2 settings	Response PDU, Response available and command accepted, ACn command

Table 14.13 shows an example of reading the RTM data.

Table 14.13

Presentation - Response frame contents example

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1101 0000	Response PDU
7	LLC Control field	n111 0111	Response available, ACn command n bit
8	LLC Status field	0000 0000	Response available and command
9	Fragmentation header	1xxx x001	No fragmentation
10	Get.response SEQUENCE {	0111	Get response
		0	IID not present

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
	OPTION indicator	1	Attribute List present
	OPTION indicator	0	Return status not present
	OPTION indicator	0	Not used
11	EID INTEGER(0..127,...)	xxxx xxxx	Responding from the RTM application Instance. No extension,
12	AttributeList SEQUENCE OF {	0000 0001	No extension, number of attributes = 1
13	Attributes SEQUENCE { AttributeId	0000 0001	No extension, AttributeId=1 (RtmData)
14	AttributeValue CONTAINER {	0000 1010	No extension, Container Choice = 1010.
15		kkkk kkkk	RtmData
16		kkkk kkkk	
17		kkkk kkkk	
...		...	
n	}}}}	kkkk kkkk	
n+1	FCS	xxxx xxxx	Frame check sequence
n+2		xxxx xxxx	
n+3	Flag	0111 1110	End Flag

DSC_51 The REDCR then closes the connection by issuing a EVENT_REPORT, RELEASE command conforming to EN 13372, 6.2, 6.3, 6.4 and EN 12834 ,7.3.8, with no specific RTM settings. Table 14.14 shows a bit encoding example of the RELEASE command.

Table 14.14

Termination. EVENT_REPORT Release frame contents

<i>Octet #</i>	<i>Attribute/Field</i>	<i>Bits in octet</i>	<i>Description</i>
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1000 s000	The frame contains a command LPDU
7	LLC Control field	0000 0011	UI command
8	Fragmentation header	1xxx x001	No fragmentation
9	EVENT_REPORT.request SEQUENCE {	0010	EVENT_REPORT (Release)
	OPTION indicator	0	Access Credentials not present

Octet #	Attribute/Field	Bits in octet	Description
	OPTION indicator	0	Event parameter not present
	OPTION indicator	0	IID not present
	Mode BOOLEAN	0	No response expected
10	EID INTEGER (0..127,...)	0000 0000	No extension, EID = 0 (System)
11	EventType INTEGER (0..127,...) }	0000 0000	Event type 0 = Release
12	FCS	xxxx xxxx	Frame check sequence
13		xxxx xxxx	
14	Flag	0111 1110	End Flag

DSC_52 The DSRC-VU is not expected to answer to the Release command. The communication is then closed.

5.4.8 4.4104 DSRC Test transaction description

DSC_53 Full tests that include securing the data, need to be carried out as defined in ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms, by authorised persons with access to security procedures, using the normal GET command as defined above.

DSC_54 Commissioning and periodic inspection tests that require decrypting and comprehension of the decrypted data content shall be undertaken as specified in ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms and ~~Appendix~~ **Sub-appendix** 9, Type Approval List of Minimum required tests.

However, the basic DSRC communication can be tested by the command ECHO. Such tests may be required on commissioning, at periodic inspection, or otherwise to the requirement of the competent control authority or ~~Regulation (EU) No. 165/2014 (See 6 below)~~.

DSC_55 In order to effect this basic communication test, the ECHO command is issued by the REDCR during a session, i.e., after an initialisation phase has been completed successfully. The sequence of interactions is thus similar to that of an interrogation:

☞ Step 1 *The REDCR* sends a ‘beacon service table’ (BST) that includes the application identifiers (AIDs) in the service list that it supports. In the RTM applications this will simply be the service with the AID value = 2.

The *DSRC-VU* evaluates the received BST, and where it identifies that the BST is requesting Freight&Fleet (AID = 2), the *DSRC-VU* shall respond. If *the REDCR* does not offer AID=2, the *DSRC-VU* shall shut down its transaction with *the REDCR*.

☞ Step 2 *The DSRC-VU* sends a request for a private window allocation.

☞ Step 3 *The REDCR* sends a private window allocation.

☞ Step 4 *The DSRC-VU* uses the allocated private window to send its vehicle service table (VST). This VST includes a list of all the different application instantiations that this *DSRC-VU* supports in the framework of AID=2. The different instantiations shall be identified by means of uniquely EIDs, each associated with a parameter value indicating the instance of the application that is supported.

☞ Step 5 Next *the REDCR* analyses the offered VST, and either terminates the connection (RELEASE) since it is not interested in anything the VST has to offer (i.e., it is receiving a VST from a *DSRC-VU* that is not an RTM VU, or, if it receives an appropriate VST it starts an app instantiation.

☞ Step 6 *The REDCR* shall issue a command (ECHO) to the specific *DSRC-VU*, and allocates a private window.

↪ Step 7 The *DSRC-VU* uses the newly allocated private window to send an ECHO response frame.

The following tables give a practical example of an ECHO exchange session.

DSC_56 Initialisation is performed according to 5.4.7 (DSC_44 – DSC_48) and Tables 14.4 – 14.9

DSC_57 The REDCR then issues an ACTION, ECHO command conforming to ISO 14906, containing 100 octets of data and with no specific settings for RTM. Table 14.15 shows the contents of the frame sent by the REDCR.

Table 14.15

ACTION, ECHO request frame example

Octet #	Attribute/Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific DSRC-VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1010 s000	Command PDU
7	LLC Control field	n111 0111	Polled ACn command, n bit
8	Fragmentation header	1xxx x001	No fragmentation
9	ACTION.request SEQUENCE {	0000	Action request (ECHO)
	OPTION indicator	0	Access Credentials not present
	OPTION indicator	1	Action parameter present
	OPTION indicator	0	IID not present
	Mode BOOLEAN	1	Response expected
10	EID INTEGER (0..127,...)	0000 0000	No extension, EID = 0 (System)
11	ActionType INTEGER (0..127,...)	0000 1111	No extension, Action type ECHO request
12	ActionParameter CONTAINER {	0000 0010	No extension, Container Choice = 2
13		0110 0100	No extension. String length = 100 octets
14		xxxx xxxx	Data to be echoed
...		...	
113	}}	xxxx xxxx	
114	FCS	xxxx xxxx	Frame check sequence
115		xxxx xxxx	
116	Flag	0111 1110	End Flag

DSC_58 The *DSRC-VU*, when receiving the ECHO request, sends an ECHO response of 100 octets of data by reflecting the received command, according to ISO 14906, with no specific settings for RTM. Table 14.16 shows a bit level encoding example.

Table 14.16
ACTION, ECHO response frame example

Octet #	Attribute/Field	Bits in octet	Description
1	FLAG	0111 1110	Start flag
2	Private LID	xxxx xxxx	Link address of the specific VU
3		xxxx xxxx	
4		xxxx xxxx	
5		xxxx xxxx	
6	MAC Control field	1101 0000	Response PDU
7	LLC Control field	n111 0111	ACn command n bit
8	LLC status field	0000 0000	Response available
9	Fragmentation header	1xxx x001	No fragmentation
10	ACTION.response SEQUENCE {	0001	ACTION response (ECHO)
	OPTION indicator	0	IID not present
	OPTION indicator	1	Response parameter present
	OPTION indicator	0	Return status not present
	Fill BIT STRING (SIZE (1))	0	Not used
11	EID INTEGER (0..127,...)	0000 0000	No extension, EID = 0 (System)
12	ResponseParameter CONTAINER {	0000 0010	No extension, Container Choice =
13		0110 0100	No extension. String length = 100
14		xxxx xxxx	Echoed data
...		
113	}}	xxxx xxxx	
114	FCS	xxxx xxxx	Frame check sequence
115		xxxx xxxx	
116	Flag	0111 1110	End Flag

5.5. ~~4.5 Support for Directive 2015/71/EC (Text deleted)~~ **Reserved for future use**

5.6. ~~4.5.7 Data transfer mechanisms between the DSRC_VU and VU~~

~~DSC_64 The Data transfer mechanisms for OWS data between the Interrogator and DSRC facility in the vehicle shall be the same as for RTM data (see 4.4.102).~~

~~DSC_65 The Data transfer between the platform collecting the maximal weights data and the DSRC facility in the vehicle shall be based on the physical connection and interfaces and protocol defined in section 4.6.~~

~~4.6 Data transfer between the DSRC.~~

5.6.1 ~~4.6.1 Physical Connection and interfaces~~

~~DSC_66 The connection between the VU and the DSRC-VU can be either by physical cable or short range wireless communication based on Bluetooth v4.0 BLE.~~

DSC_67 Regardless of the choice of the physical connection and interface, the following requirements shall be satisfied:

DSC_68 a) In order that different suppliers may be contracted to supply the VU and the DSRC-VU, and indeed different batches of DSRC-VU, the connection between the VU and the DSRC-VU not internal to the VU shall be an open standard connection. The VU shall connect with the DSRC-VU either

- (i) using fixed cable of at least 2 meters, using a Straight DIN 41612 H11 Connector – 11 pin approved male connector from the DSRC-VU to match a similar DIN/ISO approved female connector from the VU device,
- (ii) using Bluetooth Low Energy (BLE)
- (iii) using a standard ISO 11898 or SAE J1939 connection

DSC_69 b) the definition of the interfaces and connection between the VU and DSRC-VU must support the application protocol commands defined in 5.6.2. and

DSC_70 c) the VU and DSRC-VU must support the operation of the data transfer via the connection in regard to performance and power supply.

5.6.2 ~~4.6.2~~ Application Protocol

DSC_71 The application protocol between the VU Remote Communication facility and DSRC-VU is responsible for periodically transferring the remote communication data from the VU to the DSRC.

DSC_72 The following main commands are identified:

1. Initialisation of the communication link - Request
2. Initialisation of the communication link – Response
3. Send Data with Identifier of the RTM application and Payload defined by RTM Data
4. Acknowledgment of the data
5. Termination of the communication link - Request
6. Termination of the communication link – Response

DSC_73 In ASN1.0, the previous commands may be defined as:

Remote Communication DT Protocol DEFINITIONS ::= BEGIN

```

RCDT-CommunicationLink Initialization - Request ::= SEQUENCE { LinkIdentifier INTEGER
}

RCDT-CommunicationLink Initialization - Response ::= SEQUENCE { LinkIdentifier INTEGER,
    answer    BOOLEAN
}

RCDT- Send Data ::= SEQUENCE {
    LinkIdentifier          INTEGER,
    DataTransactionId     INTEGER, RCDTData
    SignedTachographPayload
}

RCDT Data Acknowledgment ::= SEQUENCE {
    LinkIdentifier          INTEGER, DataTransactionId
    INTEGER,
    answer    BOOLEAN
}

```

```

RCDT-CommunicationLink Termination - Request ::= SEQUENCE { LinkIdentifier INTEGER
}

RCDT-Communication Link Termination - Response ::= SEQUENCE { LinkIdentifier
INTEGER,
answer      BOOLEAN
}

End

```

DSC_74 The description of the commands and parameters is following:

RCDT-Communication Link Initialization – Request is used to initialize the communication link. The command is sent by the VU to the DSRC-VU. The LinkIdentifier is set by the VU and communicated to the DSRC-VU to track a specific communication link.

(Note: this is to support future links and other application/modules like Weighing on board).

RCDT-Communication Link Initialization – Response is used by the DSRC-VU to provide the response of the request to initialize the communication link. The command is sent by the DSRC-VU to the VU. The command provides the result of the initialisation as answer = 1 (Success) or =0 (Failure).

DSC_75 The initialization of the communication link shall be done only after installation, calibration, and start of the engine/**VU is switched on**.

RCDT-Send Data is used by the VU to send the signed RCDTData (i.e., *the remote communication Data*) to the DSRC-VU. The data will be sent every 60 seconds. The DataTransactionId parameter identifies the specific transmission of data. The LinkIdentifier is also used to ensure that the appropriate link is correct.

RCDT-Data Acknowledgment is sent by the DSRC-VU to provide the feedback to the VU on the reception of the data from a RCDT-Send Data command identified by the DataTransactionId parameter. The Answer parameter is 1 (Success) or =0 (Failure). If a VU receives more than three answers equal to 0 or if the VU does not receive a RCDT Data Acknowledgment for a specific previously sent RCDT- Send Data with a specific DataTransactionId, the VU will generate and record an event.

RCDT-Communication Link Termination request is sent by the VU to DSRC-VU to terminate a link for a specific LinkIdentifier.

DSC_76 At the restart of the DSRC-VU or a VU, all the existing Communication Links should be removed as there could be “dangling” Links due to the sudden shutdown of a VU.

- RCDT-Communication Link Termination – Response is sent by the DSRC-VU to the VU to confirm the request of termination of the link by the VU for the specific LinkIdentifier.

5.7. 4.7 Error handling

5.7.1 4.7.1 Recording and communication of the Data in the DSRC-VU

DSC_77 *The Data* shall be provided, already secured, by the *VUSM* function to the *DSRC-VU*. The *VUSM* shall verify that data recorded in the *DSRC-VU* has been **transmitted successfully to the DSRC-VU** recorded correctly. The recording and reporting of any errors in the transfer of data from the *VU* to the memory of the *DSRC-VU* shall be recorded with type EventFaultType and enum value set to 62-Remote-OC'H Communication Facility error **with the remote communication fault facility event** together with the timestamp. **The VUSM shall verify that the data has been transmitted successfully to the DSRC-VU.**

DSC_78 **Reserved for future use**~~The VU shall maintain a file identified by a unique name that is easily identifiable by inspectors for the purpose of recording “VU internal communication failures”.~~

DSC_79 If the VUPM attempts to obtain VU data from the security module (to pass to the VU-DSRC), but fails to do so, it shall record that failure with type EventFaultType and enum value set to ‘62’ *H Remote Communication Facility* communication fault together with the timestamp. The failure of the communication is detected when a RCDT Data Acknowledgment message is not received for the related (i.e., with the same DataTransactionId in the Send Data and Acknowledgment messages) RCDT Send Data for more than three consecutive times.

5.7.2 ~~4.7.2~~ Wireless Communication errors

DSC_80 Communication error handling shall be consistent with the related DSRC standards, namely EN 300 674-1, EN 12253, EN 12795, EN 12834 and the appropriate parameters of EN 13372.

5.7.2.1 ~~4.7.2.1~~ Encryption and signature errors

DSC_81 Encryption and signature errors shall be handled as defined in ~~Appendix~~ **Sub-appendix** 11 Common Security Mechanisms and are not present in any error messages associated with the DSRC transfer of data.

5.7.2.2 ~~4.7.2.2~~ Recording of errors

The DSRC medium is a dynamic wireless communication in an environment of uncertain atmospheric and interference conditions, particularly in the ‘portable REDCR and ‘moving vehicle’ combinations involved in this application. It is therefore necessary to ascertain the difference between a ‘read failure’ and an ‘error’ condition. In a transaction across a wireless interface, read failure is common and the consequence is usually to retry, i.e. rebroadcast the BST and reattempt the sequence, which will in most circumstances lead to a successful communication connection and transfer of data, unless the target vehicle moves out of range during the time required to retransmit. (A ‘successful’ instance of a ‘read’ may have involved several attempts and retries).

Read failure may be because the antennas were not paired properly (failure of ‘aiming’); because one of the antennas is shielded – this may be deliberate, but also can be caused by the physical presence of another vehicle; radio interference, especially from circa 5.8 GHz WIFI or other public access wireless communications, or may be caused by radar interference, or difficult atmospheric conditions (e.g. during a thunderstorm); or simply by moving out of the range of the DSRC communication. Individual instances of read failures, by their nature, cannot be recorded, simply because the communication simply did not occur.

However, if the agent of the competent control authority targets a vehicle and attempts to interrogate its *DSRC-VU*, but no successful transfer of data ensues, this failure could have occurred because of deliberate tampering, and therefore the agent of the competent control authority needs a means to log the failure, and alert colleagues downstream that there may be a violation. The colleagues can then stop the vehicle and carry out a physical inspection. However, as no successful communication has taken place, the *DSRC-VU* cannot provide data concerning the failure. Such reporting shall therefore be a function of REDCR equipment design.

‘Failure to read’ is technically different to an ‘error’. In this context an ‘error’ is the acquisition of a wrong value.

Data transferred to the *DSRC-VU* is supplied already secured, therefore must be verified by the supplier of the data (see 5.4).

Data subsequently transferred across the air interface is checked by cyclic redundancy checks at the communications level. If the CRC validates, then the data is correct. If the CRC does not validate, the data is retransmitted. The probability that data could successfully pass through a CRC incorrectly is statistically so highly improbable that it may be discounted.

If the CRC does not validate and there is no time to retransmit and receive the correct data, then the result will not be an error, but an instantiation of a specific type of read failure.

The only meaningful ‘failure’ data that can be recorded is that of the number of successful initiations of transactions that occur, that do not result in a successful transfer of data to the REDCR.

DSC_82 The REDCR shall therefore record, time-stamped, the number of occasions where the ‘initialisation’ phase of a DSRC interrogation is successful, but the transaction terminated before *the Data* was successfully retrieved by the REDCR. This data shall be available to agent of the competent control authority and shall be stored in the memory of the REDCR equipment. The means by which this is achieved shall be a matter of product design or the specification of a competent control authority.

The only meaningful ‘error’ data that can be recorded is the number of occasions where the REDCR fails to decrypt *the Data* received. However, it should be noted that this will only relate to the efficiency of the REDCR software. Data may be technically decrypted, but make no semantic sense.

DSC_83 The REDCR shall therefore record, time-stamped, the number of occasions where it has attempted but failed to decipher data received across the DSRC interface.

6. ~~5.~~ Commissioning and periodic inspection tests for the remote communication function

6.1. ~~5.1~~ General

DSC_84 Two type of tests are foreseen for the remote communication function:

- (1) An ECHO test to validate the *DSRC-REDCR >>:-<DSRC-VU wireless* communication channel.
- (2) A End-to-end security test to ensure that a workshop card is able to access the encrypted and signed data content created by the VU and transmitted over the wireless communication channel.

6.2. ~~5.2~~ ECHO

This clause contains provisions specifically made to test only that the *DSRC-REDCR >>:-<DSRC-VU* is functionally active.

The objective of the ECHO command is to enable workshops or type approval test facilities to test that the DSRC link is working without needing access to security credentials. The tester’s equipment therefore only needs to be able to initialise a DSRC communication (sending a BST with AID=2) and then send the ECHO command, and, assuming the DSRC is working, will receive the ECHO response. See 5.4.8 for details. Assuming it receives this response correctly, the DSRC link (*DSRC-REDCR >>:-<DSRC-VU*) may be validated as functioning correctly.

6.3. ~~5.3~~ Tests to validate the secure data content

DSC_85 This test is execute to validate the end-to-end security flow of data. A DSRC test reader is needed for such test. The DSRC test reader performs the same functionality and it is implemented with the same specifications of the reader used by the law enforcers, with the difference that a workshop card shall be used to authenticate the user of the DSRC test reader rather than a control card. The test can be executed after the initial activation of a Smart Tachograph or at the end of the calibration procedure. After the activation, the vehicle unit shall generate and communicate to the DSRC-VU the secured early detection data.

DSC_86 The workshop personnel must position the DSRC test reader at a distance between 2 and 10 metres in front of the vehicle.

DSC_87 Then the workshop personnel will insert a workshop card in the DSRC test reader to request the interrogation of the early detection data to the vehicle unit. After a successful interrogation, the workshop personnel will access the received data to ensure that it has been successfully validated for integrity and decrypted.

ANNEX

Rules for the computation of daily, weekly and fortnightly driving time

1. Basic computation rules

The VU shall compute the daily driving time, the weekly driving time and the fortnightly driving time using relevant data stored in a driver (or workshop) card inserted in the driver slot (slot 1, card reader #1) of the Vehicle Unit, and selected driver's activities while this card is inserted in the VU.

The driving times shall not be calculated while no driver (or workshop) card is inserted.

UNKNOWN period(s) found during the time period needed for computations shall be assimilated to BREAK/REST.

UNKNOWN periods and activities of negative duration (i.e. start of the activity occurs later than the end of the activity) due to time overlaps between two different VUs or due to time adjustment, are not taken into account.

Activities recorded in the driver card corresponding to 'OUT OF SCOPE' periods in accordance with definition (gg) of Appendix 1C, shall be interpreted as follows:

- **BREAK/REST shall be computed as 'BREAK' or 'REST'**
- **WORK and DRIVING shall be considered as 'WORK'**
- **AVAILABILITY shall be considered as 'AVAILABILITY'**

In the context of this annex, the VU shall assume to have a daily rest period at the beginning of the card activities records.

2. Concepts

The following concepts apply exclusively to this appendix, and are intended to specify the computation of driving times by the VU and its later transmission by the remote communication facility.

(a) 'RTM-shift' is the period between the end of a daily rest period and the end of the directly following daily rest period.

The VU shall start a new RTM-shift after a daily rest period has finished.

The ongoing RTM-shift is the period since the end of last daily rest period;

(b) 'accumulated driving time' is the sum of the duration of all DRIVING activities of the driver within a period while not in OUT OF SCOPE;

(c) 'daily driving time' is the accumulated driving time within a RTM-shift;

(d) 'weekly driving time' is the accumulated driving time for the ongoing week;

(e) 'continuous rest period' is any uninterrupted period of BREAK/REST;

(f) 'fortnightly driving time' is the accumulated driving time for the previous and the ongoing week;

(g) 'daily rest period' is a period of BREAK/REST, which can be either

- **a regular daily rest period,**
- **a split daily rest period or**
- **a reduced daily rest period**

In the context of Appendix 14, when a VU is computing weekly rest periods, those weekly rest periods shall be considered as daily rest periods;

(h) ‘regular daily rest period’ is a continuous rest period of at least 11 hours.

As a matter of exception, when a FERRY/TRAIN CROSSING condition is active the regular daily rest period may be interrupted a maximum of two times by activities other than rest, with a maximal accumulated duration of one hour, i.e. the regular daily rest period containing ferry/train crossing period(s) may be split into two or three parts. The VU shall then compute a regular daily rest period when the accumulated rest time computed according to point 3 is at least 11 hours.

When a regular daily rest period has been interrupted the VU:

- shall not incorporate the driving activity encountered during those interruptions to the computation of the daily driving time, and
- shall start a new RTM-shift at the end of the regular daily rest period that has been interrupted.

		⚠				⚠			
A	B	C	D	E	F	G			
⓪/✖/ⓧ	h	⓪/✖/ⓧ	h ⚠	⓪/✖/ⓧ	h	⓪/✖/ⓧ			
Working Period	2 h	30 min	8 h	30 min	2 h	New Day			

Figure 1. Example of daily rest period interrupted due to ferry/train crossing

- (i) ‘reduced daily rest period’ is a continuous rest period of at least 9 hours and less than 11 hours;
- (j) ‘split daily rest period’ is a daily rest period taken in two parts:
 - the first part shall be a continuous rest period of at least 3 hours and less than 9,
 - the second part shall be a continuous rest period of at least 9 hours.

As a matter of exception, when a FERRY/TRAIN CROSSING condition is active during one or both parts of a split daily rest period, the split daily rest period may be interrupted a maximum of two times by other activities with the accumulated duration of maximal one hour, i.e.:

- the first part of the split daily rest period may be interrupted one or two times, or
- the second part of the split daily rest period may be interrupted one or two times, or
- the first part of the split daily rest period may be interrupted one time and the second part of the split daily rest period may be interrupted one time.

The VU shall then compute a split daily rest period when the accumulated rest time computed according to point 3 is:

- at least three hours and less than 11 hours for the first rest period and at least 9 hours for the second rest period, when the first rest period has been interrupted by FERRY/TRAIN CROSSING.
- at least three hours and less than 9 hours for the first rest period and at least 9 hours for the second rest period, when the first rest period has not been interrupted by FERRY/TRAIN CROSSING.

		⚠		⚠		⚠		⚠			
A	B	C	D	E	F	G	H	I			
⓪/✖/ⓧ	h	⓪/✖/ⓧ	h ⚠	⓪/✖/ⓧ/h	h ⚠	⓪/✖/ⓧ	h	⓪/✖/ⓧ			
4 h	1h	20 min	2 h	6 h	7h	20 min	3h	New Day			

Figure 2. Example of split daily rest period interrupted due to ferry/train crossing

When the split daily rest period is interrupted, the VU:

- shall not incorporate the driving activity encountered during those interruptions to the computation of the daily driving time, and
- shall start a new RTM-shift at the end of the split daily rest period that has been interrupted;

(k) 'week' is the period in UTC time between 00:00 hours on Monday and 24:00 hours on Sunday;

3. Computation of the rest period when it has been interrupted due to ferry/train crossing

For the computation of the rest period when it has been interrupted due to ferry/train crossing, the VU shall calculate the accumulated rest time according to the following steps:

a) Step 1

The VU shall detect interruptions to the rest time occurring before the activation of the FERRY/TRAIN CROSSING (BEGIN) flag, according to figure 3 and in its case figure 4, and shall evaluate for each interruption detected if the following conditions are met:

- the interruption makes the total duration of the interruptions detected, including in its case interruptions occurring during the first part of a split daily rest period due to ferry/train crossing, to exceed more than one hour in total,
- the interruption makes the total number of interruptions detected, including in its case interruptions occurring during the first part of a split daily rest period due to ferry/train crossing, to be bigger than two,
- there is an 'Entry of place where daily work periods end' stored after the interruption ended.

If none of the above conditions are met, the continuous rest period immediately preceding the interruption shall be added to the accumulated rest time.

If at least one of the above conditions is met, the VU shall either stop the computation of the accumulated rest time according to step 2 or detect interruptions to the rest time occurring after the FERRY/TRAIN CROSSING (BEGIN) flag according to step 3.

b) Step 2

For each interruption detected according to step 1, the VU shall evaluate whether the computation of the accumulated rest time should stop. The VU shall stop the computation process when two continuous rest periods occurring before the activation of the FERRY/TRAIN CROSSING (BEGIN) flag have been added to the accumulated rest time, including in its case rest periods added in the first part of a split daily rest period also interrupted by ferry/train crossing. Otherwise, the VU shall proceed according to step 3.

c) Step 3

If after performance of step 2 the VU continues the computation of the accumulated rest time, the VU shall detect interruptions occurring after the deactivation of the FERRY/TRAIN CROSSING condition according to figure 3 and in its case figure 4.

For each interruption found, the VU shall evaluate if the interruption makes the accumulated time of all the interruptions detected to exceed more than one hour in total, in which case the computation of the accumulated rest period shall finish at the end of the continuous rest period previous to the interruption. Otherwise, the continuous rest periods occurring after the respective interruptions shall be added to the computation of the daily rest period until the condition in step 4 is fulfilled.

d) Step 4

The computation of the accumulated rest time shall stop when the VU has added, as result of steps 1 and 3, a maximum of two continuous rest periods to the rest period for which the FERRY/TRAIN

CROSSING condition is activated, including in its case interruptions occurring during the first part of a split daily rest period due to ferry/train crossing.

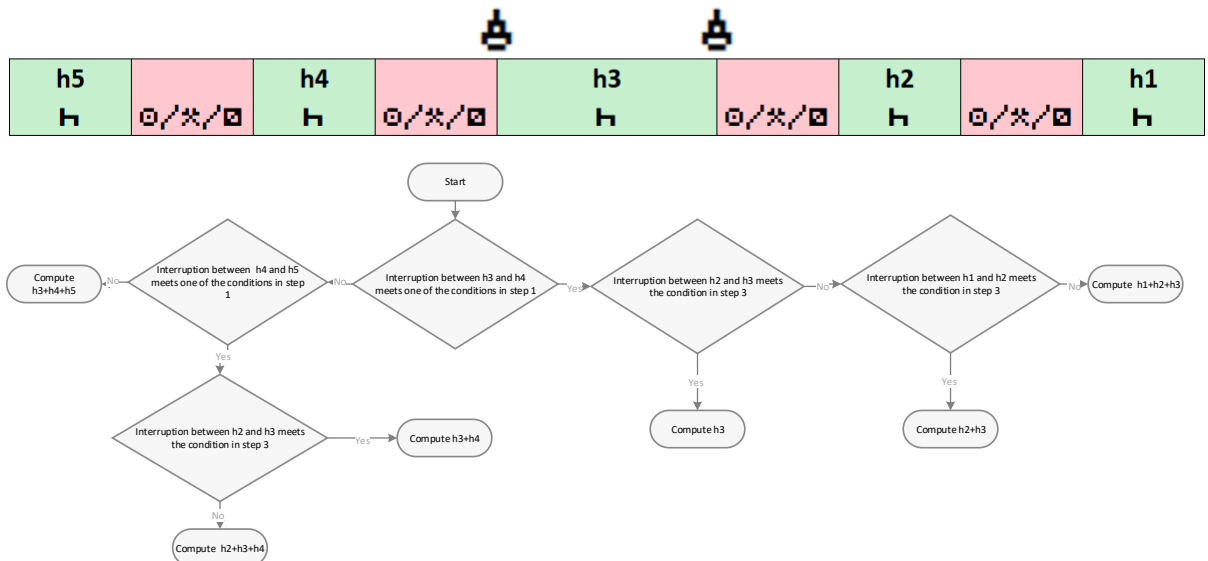


Figure 3. Processing of rest times by the VU in order to determine whether an interrupted rest period shall compute as regular daily rest period or as the first part of a split daily rest period

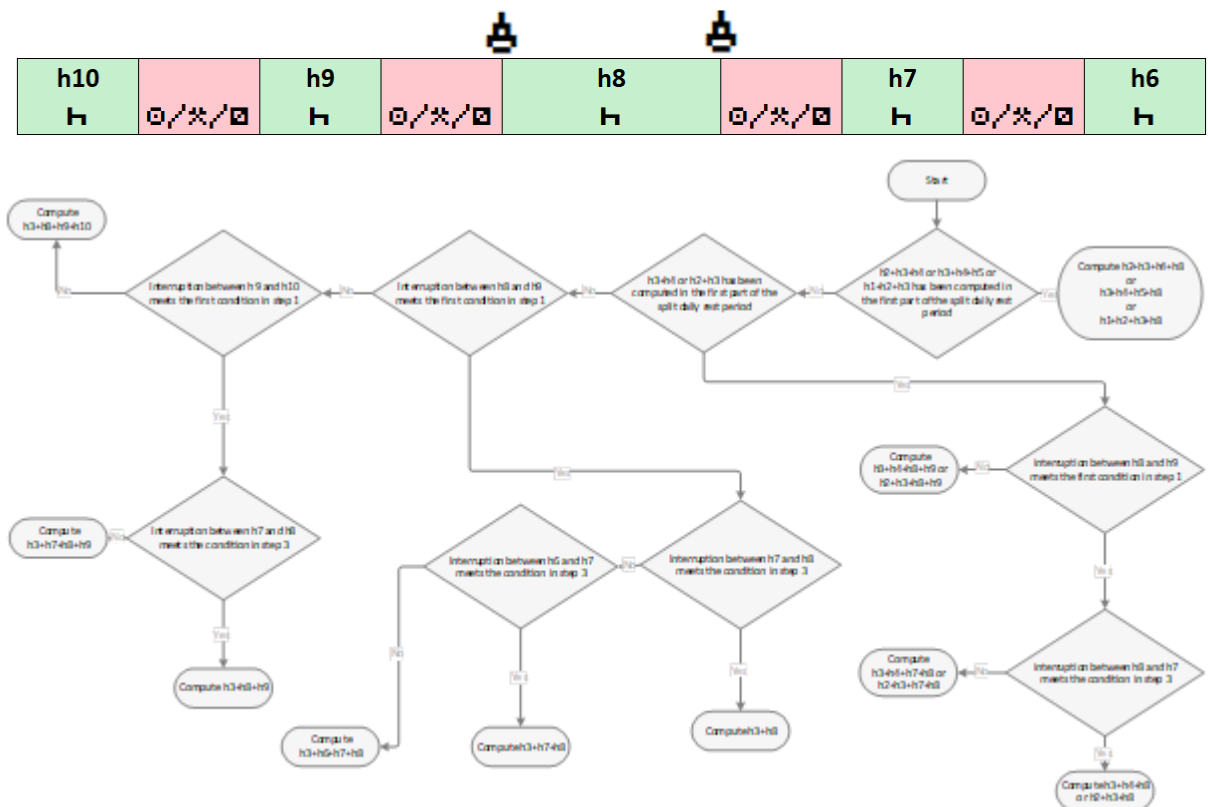


Figure 4. Processing of rest times by the VU in order to determine whether an interrupted rest period shall compute as the second part of a split daily rest period

A	B	C	D	E	F	G	H	I
⊙/*/⊠/⊠/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠/⊠
4,5h	2h	20 min	1h	20 min	8h	20 min	2h	
Working	Rest	Movement	Rest	Embarking	Rest on Ferry	Disembark	Rest	
						Start of new shift due to three interruptions		Manually selected Start of new work period

Figure 5. Example of a Daily rest period interrupted more than twice causing rest period H not to be included in the computation

A	B	C	D	E	F	G	H	I
⊙/*/⊠/⊠/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠/⊠
4,5h	2h	20 min	1h	20 min	8h	20 min	2h	
Working	Rest	Movement	Rest	Embarking	Rest on Ferry	Disembark	Rest	
			Manually selected End of work period					Manually selected Start of new work period
								Start of New Shift

Figure 6. Example of a Daily rest period where Ferry/Train Calculation period is commenced at End of Work Period

A	B	C	D	E	F	G	H	I
⊙/*/⊠/⊠/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠/⊠
4,5h	1h	10 min	1h	10 min	1h	10 min	9h	
Working	Rest	Movement	Rest	Movement	Rest	Embarking	Rest on ferry	
								Start of New Shift

Figure 7. Example of a Daily rest period interrupted more than twice causing rest period B not to be included in the computation

A	B	C	D	E	F	G	H	I
⊙/*/⊠/⊠/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠/⊠	⊠	⊙/*/⊠	⊠	⊙/*/⊠/⊠
3h	1h	10 min	2h	6h	2h	10 min	7h	
Working	Rest	Embarking	Rest on ferry	Working	Rest	Embarking	Rest on ferry	
								Start of New Shift

Figure 8. Example of a Split Daily rest period interrupted once during the first Rest Period and once during 2nd Rest Period

4. Computation of daily, weekly and fortnightly- driving times

The VU shall compute the daily driving time(s) for the ongoing and previous RTM-shifts. The driving time occurring during the interruptions of the daily rest periods shall not be added to the computation of the daily driving time, when such interruptions are due to ferry/train crossing and the requirements provided for in paragraphs (h) and (j) of point 2 and in point 3 have been fulfilled. Nevertheless, insofar as a complete regular or split daily rest period has not been computed by the VU according to point 3, the driving times occurring during the interruptions shall be added to the daily driving time for the ongoing RTM-shift.

The VU shall also compute the weekly and the fortnightly driving times. The driving time occurring during the interruptions of the daily rest periods due to ferry/train crossing shall be added to the computation of the weekly and the fortnightly driving times.

Sub-appendix 15. Migration: Managing the co-existence of equipment generations

TABLE OF CONTENT

1.	DEFINITIONS.....	608
2.	GENERAL PROVISIONS.....	608
2.1.	<i>Overview of the transition</i>	609
2.2.	<i>Interoperability between VU and cards.....</i>	609
2.3.	<i>Interoperability between VU and MS.....</i>	609
2.4.	<i>Interoperability between vehicle units, tachograph cards and equipment for data download.....</i>	609
2.4.1	<i>Direct card download by IDE.....</i>	610
2.4.2	<i>Card download through a vehicle unit.....</i>	610
2.4.3	<i>Vehicle unit download.....</i>	610
2.5.	<i>Interoperability between VU and calibration equipment</i>	610
3.	MAIN STEPS DURING THE PERIOD BEFORE THE INTRODUCTION DATE	610
4.	PROVISIONS FOR THE PERIOD AFTER THE INTRODUCTION DATE.....	611

1. Definitions

For the purposes of this ~~Appendix~~ **sub-appendix**, the following definitions are used.

smart tachograph system: as defined by this ~~Annex~~ **Appendix** (chapter 1: definition bbb);

first generation tachograph system: as defined ~~by~~ **in the introduction of this Regulation** (~~Article 2: definition 1~~) **Appendix**;

second generation tachograph system: as defined ~~by~~ **in the introduction of this Regulation** (~~Article 2: definition 1~~) **Appendix**;

introduction date: as defined by this ~~Annex~~ **Appendix** (chapter 1: definition ccc);

Intelligent Dedicated Equipment (IDE): equipment used to perform data downloading, as defined in ~~Appendix~~ **Sub-appendix 7 of this Annex** **Appendix**.

2. General Provisions

2.1. Overview of the transition

The ~~introduction~~ **preamble** of this ~~Annex~~ **Sub-appendix** provides an overview of the transition between the first and the second generation tachograph systems, **and of the introduction of the second version of second generation recording equipment and tachograph cards.**

In addition to the provisions of this **introduction**, **the following information can be reminded** ~~preamble~~:

- first generation motion sensors ~~are~~ **will** not be interoperable with **any version of** second generation vehicle units,,-
- **only** second generation motion sensors ~~can~~ **will start to** be installed in vehicles **equipped with any version of** ~~at the same time as~~ second generation vehicle units,,-
- data download and calibration equipment ~~will~~ **need to evolve, in order to** support use of both generations **or versions of recording equipment control devices** and tachograph cards.

2.2. Interoperability between VU and cards

It is understood that first generation tachograph cards are interoperable with first generation vehicle units (in compliance with ~~Annex 1 B of this Regulation~~) **Appendix IB, any version of** ~~while~~ second generation tachograph cards are interoperable with **any version of** second generation vehicle units in compliance with ~~Annex Appendix 1C of this Regulation~~) **Agreement**. In addition, the requirements below shall apply.

MIG_001 Except as provided for in requirement MIG_004 and MIG_005, first generation tachograph cards may continue to be used in **any version of** second generation vehicle units until their end of validity date. Their holders may however ask for their replacement by second generation tachograph cards as soon as they are available.

MIG_002 **Any version of** ~~S~~second generation vehicle units shall be able to use any valid first generation driver, control and company card inserted.

MIG_003 This capability may be suppressed once and forever in such vehicle units by workshops, so that first generation tachograph cards cannot be accepted anymore. This may

only be done after ~~the European Commission has launched~~ a procedure aiming to request workshops to do so ~~has been launched~~, for example during each periodic inspection of tachograph.

MIG_004 Second generation vehicle units shall only be able to use second generation workshop cards.

MIG_005 For determining the mode of operation, **any version of** second generation vehicle units shall only consider the types of the valid cards inserted, regardless of their generations **or versions**.

MIG_006 Any **version of** valid second generation tachograph card shall be able to be used in first generation vehicle units exactly the same manner as a first generation tachograph card of the same type.

2.3. Interoperability between VU and MS

It is understood that first generation motion sensors are interoperable with first generation vehicle units, while second generation motion sensors are interoperable with **any version of** second generation vehicle units. In addition, the requirements below shall apply.

MIG_007 **Any version of** second generation vehicle units ~~wi~~shall not be able to be paired and used with first generation motion sensors.

MIG_008 Second generation motion sensors may be paired and used with second generation vehicle units only, **whichever the version**, or with both generations of vehicle units.

2.4. Interoperability between vehicle units, tachograph cards and equipment for data download

MIG_009 Equipment for data download may be ~~compatible used with one all generations and versions only~~ of vehicle units and tachograph cards, ~~or with both~~.

2.4.1 Direct card download by IDE

MIG_010 Data shall be downloaded by IDE from tachograph cards of one generation inserted in their card readers, using the security mechanisms and the data download protocol of this generation, and downloaded data shall have the format defined for this generation **and version**.

MIG_011 To allow drivers' control by non EU control authorities, it shall also be possible to download second generation driver (and workshop) cards, **whichever the version, in** exactly the same manner as **first**^{1st} generation drivers (and workshop) cards. Such download shall include:

- non signed EFs IC and ICC (**optional**),
- non signed EFs (4th-**first** generation) Card_Certificate and CA_Certificate,
- the other application data EFs (within DF **Tachograph**) requested by the first generation card download protocol. This information shall be secured with a digital signature, according to the first generation security mechanisms.

Such download shall not include application data EFs only present in **version 1 and version 2** second generation driver (and workshop) cards (application data EFs within **DF Tachograph_G2**).

2.4.2 Card download through a vehicle unit

MIG_012 Data shall be downloaded from **any version of** second generation card, inserted in a first generation vehicle unit using the first generation data download protocol. The card shall answer to the vehicle unit commands exactly the same manner as a first generation card and downloaded data shall have the same format as data downloaded from a first generation card.

MIG_013 Data shall be downloaded from a first generation card inserted in **any version of** second generation vehicle unit using the data download protocol defined in ~~Appendix Sub-**appendix** 7 of this Annex~~ **Appendix**. The vehicle unit shall send commands to the card exactly the same manner as a first generation vehicle unit, and downloaded data shall respect the format defined for first generation cards.

2.4.3 Vehicle unit download

MIG_014 ~~Data~~ **Outside of the frame of drivers' control by non EU control authorities, data** shall be downloaded from second generation vehicle units using the second generation security mechanisms, and the data download protocol specified in ~~Appendix Sub-**appendix** 7 of this Annex~~ **Appendix for the relevant version**.

MIG_015 To allow drivers' control by non EU control authorities, it may optionally also be possible to download data from **any version of** second generation vehicle units using the first generation security mechanisms. Downloaded data shall **then** have the same format as data downloaded from a first generation vehicle unit. This capability may be selected through commands in the menu.

2.5. Interoperability between VU and calibration equipment

MIG_016 Calibration equipment shall be able to perform calibration of each generation **or version** of tachograph, using the calibration protocol of this generation. Calibration equipment may be ~~compatible used~~ **with all one generations and versions of vehicle units only of tachograph, or with both**.

3. Main steps during the period before the introduction date

MIG_017 Test keys and certificates shall be available to manufacturers at the ~~latest 30 months before the introduction~~ **publication date of this appendix**.

MIG_018 Interoperability tests shall be ready to start **with version 2 of vehicle units and version 2 of tachograph cards** if requested by manufacturers at the latest 15 months before the introduction date.

MIG_019 **For version 2 of generation 2 tachographs, tachograph cards and motion sensors, the same** ~~Official~~ **keys and certificates are used as for generation 2 version 1 equipments** ~~shall be available to manufacturers at the latest 12 months before the introduction date~~.

MIG_020 ~~Member States~~ **Contracting Parties** shall be able to issue **version 2 of** second generation workshop cards at the latest ~~3-1~~ **month** before the introduction date.

MIG_021 ~~Member States~~ **Contracting Parties** shall be able to issue all **other** types of **version 2 of** second generation tachograph cards at the latest **1 month before the introduction date**.

4. Provisions for the period after the introduction date

MIG_022 **With effect from**~~After~~ the introduction date, ~~Member States Contracting Parties~~ shall only issue **version 2 of** second generation tachograph cards.

MIG_023 Vehicle units / motion sensors manufacturers shall be allowed to produce first generation vehicle units / motion sensors as long as they are used in the field, so that malfunctioning components can be replaced.

MIG_024 With effect from the introduction date, malfunctioning version 1 of second generation vehicle units or external GNSS facilities shall be replaced with version 2 of second generation vehicle units or external GNSS facilities.

MIG_0254 Vehicle units / motion sensors manufacturers shall be allowed to request and obtain type approval maintenance of first generation vehicle units / motion sensors types **or version 1 of second generation vehicle units** already type approved.

Sub-appendix 16. Adaptor for M1 and N1 category vehicles

TABLE OF CONTENT

1. Abbreviations and reference documents.....	613
1.1. Abbreviations	613
1.2. Reference standards	613
2. General characteristics and functions of the adaptor.....	613
2.1. Adaptor general description	613
2.2. Functions	613
2.3. Security	614
3. Requirements for the Recording equipment control device when an adaptor is installed.....	614
4. Construction and functional requirements for the adaptor.....	614
4.1. Interfacing and adapting incoming speed pulses	614
4.2. Inducing the incoming pulses to the embedded motion sensor	615
4.3. Embedded motion sensor	615
4.4. Security requirements	615
4.5. Performance characteristics	616
4.6. Materials	616
4.7. Markings	616
5. Installation of the Recording equipment control device when an adaptor is used.....	616
5.1. Installation	616
5.2. Sealing	617
6. Checks, inspections and repairs.....	617
6.1. Periodic inspections	617
7. Type approval of Recording equipment control device when an adaptor is used.....	617
7.1. General points	617
7.2. Functional certificate	617

1. Abbreviations and reference documents

1.1. Abbreviations

TBD To Be Defined

VU Vehicle Unit

1.2. Reference standards

ISO16844-3 Road vehicles – Tachograph systems – Part 3: Motion sensor interface

2. General characteristics and functions of the adaptor

2.1. Adaptor general description

ADA_001 The adaptor shall provide a connected VU with secured motion data permanently representative of vehicle speed and distance travelled.

The adaptor is only intended for those vehicles that are required to be equipped with ~~recording equipment~~ **control device** in compliance with this ~~Regulation~~ **Agreement**.

It shall be installed and used only in those types of vehicle defined in definition yy) ‘adaptor’ of ~~Annex~~ **Appendix 1C** where it is not mechanically possible to install any other type of existing motion sensor which is otherwise compliant with the provisions of this ~~Annex~~ **Appendix** and its ~~Appendixes~~ **Sub-appendixes** 1 to 16.

The adaptor shall not be mechanically interfaced to a moving part of the vehicle, but connected to the speed/distance impulses which are generated by integrated sensors or alternative interfaces.

ADA_002 A type approved motion sensor (according to the provisions of this ~~Annex~~ **Appendix 1C**, section 8, Type approval of ~~recording equipment~~ **control device** and tachograph cards) shall be fitted into the adaptor housing, which shall also include a pulse converter device inducing the incoming pulses to the embedded motion sensor. The embedded motion sensor itself shall be connected to the VU, so that the interface between the VU and the adaptor shall be compliant with the requirements set out in ISO16844-3.

2.2. Functions

ADA_003 The adaptor shall include the following functions:

interfacing and adapting the incoming speed pulses,

inducing the incoming pulses to the embedded motion sensor,

all functions of the embedded motion sensor, providing secured motion data to the VU.

2.3. Security

ADA_004 The adaptor shall not be security certified according to the motion sensor generic security target defined in ~~Appendix~~ **Sub-appendix 10** of this ~~Annex~~ **Appendix**.

Security related requirements specified in section 4.4 of this ~~Appendix~~ **Sub-appendix** shall apply instead.

3. Requirements for the ~~recording equipment~~ control device when an adaptor is installed

The requirements in the following Chapters indicate how the requirements of this ~~Annex~~ **Sub-appendix** shall be understood when an adaptor is used. The related requirement numbers of ~~Annex~~ **Appendix 1C** are provided between brackets.

ADA_005 The ~~recording equipment~~ control device of any vehicle fitted with an adaptor must comply with all the provisions of this ~~Annex~~ **Sub-appendix**, except otherwise specified in this ~~Appendix~~ **Sub-appendix**.

ADA_006 When an adaptor is installed, the ~~recording equipment~~ control device includes cables, the adaptor (including a motion sensor), and a VU [01].

ADA_007 The detection of events and/or faults function of the ~~recording equipment~~ control device is modified as follows:

the “power supply interruption” event shall be triggered by the VU, while not in calibration mode, in case of any interruption exceeding 200 milliseconds of the power supply of the embedded motion sensor [79]

- the “motion data error” event shall be triggered by the VU in case of interruption of the normal data flow between the embedded motion sensor and the VU and/or in case of data integrity or data authentication error during data exchange between the embedded motion sensor and the VU [83]
- the “security breach attempt” event shall be triggered by the VU for any other event affecting the security of the embedded motion sensor, while not in calibration mode [85]
- the ~~recording equipment~~ “control device” fault shall be triggered by the VU, while not in calibration mode, for any fault of the embedded motion sensor [88]

ADA_008 The adaptor faults detectable by the ~~recording equipment~~ control device shall be those related with the embedded motion sensor [88].

ADA_009 The VU calibration function shall allow to automatically pair the embedded motion sensor with the VU [202, 204].

4. Construction and functional requirements for the adaptor

4.1. Interfacing and adapting incoming speed pulses

ADA_011 The adaptor input interface shall accept frequency pulses representative of the vehicle speed and distance travelled. Electrical characteristics of the incoming pulses are: TBD by the manufacturer. Adjustments accessible to only the adaptor manufacturer, and to the approved workshop performing the adaptor installation shall allow the correct interfacing of the adaptor input to the vehicle, if applicable.

ADA_012 The adaptor input interface shall be able, if applicable, to multiply or divide the frequency pulses of the incoming speed pulses by a fixed factor, to adapt the signal to the k factor range defined by this ~~Annex~~ **Appendix** (24000 to 25000 pulses/km). This fixed

factor may only be programmed by the adaptor manufacturer, and the approved workshop performing the adaptor installation.

4.2. Inducing the incoming pulses to the embedded motion sensor

ADA_013 The incoming pulses, possibly adapted as specified above, shall be induced to the embedded motion sensor, so that each incoming pulse shall be detected by the motion sensor.

4.3. Embedded motion sensor

ADA_014 The embedded motion sensor shall be stimulated by the induced pulses, thus allowing it to generate motion data accurately representing the vehicle movement, as if it was mechanically interfaced to a moving part of the vehicle.

ADA_015 The identification data of the embedded motion sensor shall be used by the VU to identify the adaptor [95].

ADA_016 The installation data stored in the embedded motion sensor shall be considered to represent the adaptor installation data [122].

4.4. Security requirements

ADA_017 The adaptor housing shall be designed so that it cannot be opened. It shall be sealed, so that physical tampering attempts can be easily detected (e.g. through visual inspection, see ADA_035). Seals shall follow the same requirements of motion sensor seals [398 to 406]

ADA_018 It shall not be possible to remove the embedded motion sensor from the adaptor without breaking the seal(s) of the adaptor housing, or breaking the seal between the sensor and the adaptor housing (see ADA_034).

ADA_019 The adaptor shall ensure that motion data may only be processed and derived from the adaptor input.

4.5. Performance characteristics

ADA_020 The adaptor shall be fully operational in the temperature range defined by the manufacturer.

ADA_021 The adaptor shall be fully operational in the humidity range 10% to 90% [214].

ADA_022 The adaptor shall be protected against over-voltage, inversion of its power supply polarity, and short circuits [216].

ADA_023 The adaptor shall either:

react to a magnetic field disturbing vehicle motion detection. In such circumstances, the vehicle unit will record and store a sensor fault [88] or,

have a sensing element that is protected from, or immune to, magnetic fields [217].

ADA_024 The adaptor shall conform to international regulation UN ECE R10, related to electromagnetic compatibility, and shall be protected against electrostatic discharges and transients [218].

4.6. Materials

ADA_025 The adaptor shall meet the protection grade (TBD by the manufacturer, depending on the installation position) [220, 221].

ADA_026 The colour of the adaptor housing shall be yellow.

4.7 Markings

ADA_027 A descriptive plaque shall be affixed to the adaptor and shall show the following details:

- name and address of the manufacturer of the adaptor,
- manufacturer's part number and year of manufacture of the adaptor,
- approval mark of the adaptor type or of the ~~recording equipment~~ **control device** type including the adaptor,
- the date on which the adaptor has been installed,
- the vehicle identification number of the vehicle on which it has been installed.

ADA_028 The descriptive plaque shall also show the following details (if not directly readable from the outside on the embedded motion sensor):

- name of the manufacturer of the embedded motion sensor,
- manufacturer's part number and year of manufacture of the embedded motion sensor,
- approval mark for the embedded motion sensor.

5. Installation of the ~~recording equipment~~ **control device** when an adaptor is used

5.1. Installation

ADA_029 Adaptors to be installed in vehicles shall only be installed by vehicle manufacturers, or by approved workshops, authorised to install, activate and calibrate digital and smart tachographs.

ADA_030 Such approved workshop installing the adaptor shall adjust the input interface and select the division ratio of the input signal (if applicable).

ADA_031 Such approved workshop installing the adaptor shall seal the adaptor housing.

ADA_032 The adaptor shall be fitted as close as possible to that part of the vehicle which provides its incoming pulses.

ADA_033 The cables for providing the adaptor power supply shall be red (positive supply) and black (ground).

5.2. Sealing

ADA_034 The following sealing requirements shall apply:

- the adaptor housing shall be sealed (see ADA_017),

- the housing of the embedded sensor shall be sealed to the adaptor housing, unless it is not possible to remove the sensor from the adaptor housing without breaking the seal(s) of the adaptor housing (see ADA_018),
- the adaptor housing shall be sealed to the vehicle,
- the connection between the adaptor and the equipment which provides its incoming pulses shall be sealed on both ends (to the extent of what is reasonably possible).

6. Checks, inspections and repairs

6.1. Periodic inspections

ADA_035 When an adaptor is used, each periodic inspection (periodic inspections means in compliance with Requirement [409] through to Requirement [413] of ~~Annex~~ **Appendix 1C**) of the ~~recording equipment~~ **control device** shall include the following checks:

- that the adaptor carries the appropriate type approval markings,
- that the seals on the adaptor and its connections are intact,
- that the adaptor is installed as indicated on the installation plaque,
- that the adaptor is installed as specified by the adaptor and/or vehicle manufacturer,
- that mounting an adaptor is authorised for the inspected vehicle.

ADA_036 These inspections shall include a calibration and a replacement of all seals, whatever their state.

7. Type approval of ~~recording equipment~~ control device when an adaptor is used

7.1. General points

ADA_037 ~~recording equipment~~ **Control device** shall be submitted for type approval complete, with the adaptor [425].

ADA_038 Any adaptor may be submitted for its own type approval, or for type approval as a component of a ~~recording equipment~~ **control device**.

ADA_039 Such type approval shall include functional tests involving the adaptor. Positive results to each of these tests are stated by an appropriate certificate [426].

7.2. Functional certificate

ADA_040 A functional certificate of an adaptor or of ~~recording equipment~~ **control device** including an adaptor shall be delivered to the adaptor manufacturer only after all the following minimum functional tests have been successfully passed.

<i>No</i>	<i>Test</i>	<i>Description</i>	<i>Related requirements</i>
1.	Administrative examination		
1.1	Documentation	Correctness of documentation of the adaptor	
2.	Visual inspection		
2.1.	Compliance of the adaptor with documentation		
2.2.	Identification / markings of the adaptor		ADA_027, ADA_028
2.3	Materials of the adaptor		[219] to [223] ADA_026
2.4.	Sealing		ADA_017, ADA_018, ADA_034
3.	Functional tests		
3.1	Inducing the speed pulses to the embedded motion sensor		ADA_013
3.2	Interfacing and adapting incoming speed pulses		ADA_011, ADA_012
3.3	Motion measurement accuracy		[30] to [35], [217]
4.	Environmental tests		
4.1	Manufacturer test results	Results of manufacturer environment tests.	ADA_020, ADA_021, ADA_022, ADA_024
5.	EMC		
5.1	radiated emissions and susceptibility	Verify compliance with Directive UNECE 2006/28/EC-UNECE Regulation 10	ADA_024
5.2	Manufacturer test results	Results of manufacturer environment tests.	ADA_024