SISTEMAS ELECTRÓNICOS DE COBRO Y OTRAS APLICACIONES

ESPECIFICACIÓN PARA LA INTEROPERABILIDAD EN LA TRANSACCIÓN ANTENA - TRANSPONDER

MINISTERIO DE OBRAS PÚBLICAS, TRANSPORTES Y TELECOMUNICACIONES

CHILE

VERSIÓN 1.35

10 DE ENERO 2005

SISTEMAS ELECTRÓNICOS DE COBRO Y OTRAS APLICACIONES

ESPECIFICACIÓN PARA LA INTEROPERABILIDAD EN LA TRANSACCIÓN ANTENA - TRANSPONDER

- **Documento Número** : ST1
- **Versión** : 1.35
- **Estado** : Séptima versión
- **Fecha de Emisión** : 10 de Enero de 2005

- **Autores : Consultores**

Ronald Bull (Fundación Chile)
Guillermo Cuadra (IGYC, Chile)
Bernhard Oehry (RAPP, Suiza)

- **Autores : MOPTT**

Rodrigo Bravo (MOPTT, Chile)

- **Contacto** :

Rodrigo Bravo
Ministerio de Obras Públicas, Transportes y Telecomunicaciones – Chile
Tel. : +56 2-258-3722
Fax: +56 2-258-3779
Email: rodrigo.bravo@moptt.gov.cl

MINISTERIO DE OBRAS PÚBLICAS, TRANSPORTES Y TELECOMUNICACIONES
CHILE

SISTEMAS ELECTRÓNICOS DE COBRO Y OTRAS APLICACIONES

Especificación para la Interoperabilidad en la Transacción Antena - Transponder

CONTENIDO

1 Historia de Revisiones

Edición	Fecha	Observaciones
V1	8 Nov 2000	Borrador preliminar
V 1.0	5 Enero 2001	Primera versión. Cambios en la estructura, trasladando porciones a anexos. Además revisión general del texto y finalización de codificación de las transacciones.
V 1.1	1 de Julio 2001	Segunda Versión. Cambios introducidos: • En la sección 6.6 bajo la figura 6.3, se corrige referencia: "sección 5.3.2" pasa a "Anexo A, sección A.2.3.2." • En la sección 6.6 bajo la figura 6.3, se identifica en mejor forma quiénes pueden recalcular y verificar el valor de ReceiptAuthenticator. • Se corrige el texto en la sección 6.11: las claves MEAuKF no se distribuyen a las concesiones. • En figura A.2.1: Definición de EID • El uso de ReceiptAuthenticator se hace optativo a través de: ✓ Cambios en las figuras A.2.2 y A.2.3 ✓ Ajustes en el texto de A.2.3.2.2 ✓ Supresión de artículo A.2.3.2.3 ✓ Renumeración de artículos: ✓ Antiguo A.2.3.2.4 pasa a nuevo A.2.3.2.3 ✓ Antiguo A.2.3.2.5 pasa a nuevo A.2.3.2.4 ✓ Antiguo A.2.3.2.6 pasa a nuevo A.2.3.2.5 ✓ Parte de antiguo artículo A.2.3.2.3 se transfiere a nuevo artículo A.2.3.2.3. ✓ Ajuste del texto en nuevo artículo A.2.3.2.5. • La lectura optativa de Atributos del elemento de sistema se traslada de la fase de Presentación a la fase de Autenticación Fiscal, a través de: ✓ Cambios en las figuras A.2.2 y A.2.3 ✓ Ajustes en el texto del nuevo artículo A.2.3.2.3. ✓ Cambios en el Anexo C • Se reestructura el Anexo C, agregando codificación correspondiente a casos de transponder con respuesta lenta. • Se corrigen erratas de codificación de las transacciones en el Anexo C.
V 1.15	Octubre 1, 2001	Tercera Versión. Cambios introducidos: • Correcciones al texto en las secciones 4.1.4 y 6.7, y en la figura 6.2.

- SET.Request opcional agregado en la fase de Recibo de la transacción: en figura A.2.2 y en el texto de la sección A.2.3.5.
- Fase de Recibo agregada en sección A.3.
- Se corrigen otras erratas de codificación de las transacciones en el Anexo C.
- Ajuste del texto en la sección C.1.4, bytes 56 y 57, referente a StatusFlags
- En el Anexo C, el valor del número PDU sólo queda fijo durante la fase de inicialización.

V 1.2 Enero 21, 2002 Cuarta Versión. Cambios introducidos:

- "Ministerio de Obras Públicas" cambiado a "Ministerio de Obras Públicas, Transportes y Telecomunicaciones".
- Al final de la sección 3.1, se agregan referencias [MOPTT - ST2] y [MOPTT - ST3].
- En la sección 6.4, se ajusta el texto para indicar que el valor de FiscalAuthenticator no es calculado o revisado por el RSE.
- En la sección 6.11, se modifica el texto para hacer referencia a los requerimientos establecidos en [MOPTT - ST3].
- En 7.4, se agrega la condición de lista verde.
- En 7.4, 7.4.1, 7.4.2 y 7.4.3, se modifica el texto para definir quienes tienen la responsabilidad y el derecho de manipular los bits de lista negra, gris y amarilla del transponder.
- Se agrega la sección 7.4.4, correspondiente a la lista verde.
- En el capítulo 9, se modifica el texto para referirse a los requerimientos establecidos en [MOPTT - ST2].
- En B.2, ContractProvider en conjunto con el MOPTT definen el valor del byte 6 del Atributo EFC-ContextMark.
- En B2, se agrega el bit de lista verde al Atributo EquipmentStatus.

V 1.21 Enero 31, 2002 Cuarta Versión Actualizada. Cambios introducidos:

- Se corrigen erratas de codificación de las transacciones en el Anexo C.

V 1.25 Julio 15, 2002 Quinta Versión. Cambios introducidos:

- En la sección 7.4.3 se modifica el texto para hacer opcional el uso de la lista amarilla
- Se corrigen erratas en byte 26 de C.2.2 y byte 25 de C.2.3.

V 1.3 Agosto 31, 2004 Sexta Versión. Cambios introducidos:

- Se inserta nueva sección 2.3: Nuevas versiones y documentos complementarios. Se renumera, pasando antigua sección 2.3 a nueva 2.4 y antigua 2.4 a nueva 2.5.

- En nueva sección 2.4 se agregan definiciones para nativo, foráneo y Televía.
- En nueva sección 2.5 se agrega sigla AIA
- En la sección 3.1 se agrega referencia a MOPTT-ST1-1
- En la sección 4.1 y en los puntos 4.1.2, 4.1.3, 4.1.4 y 4.1.5 se cambia el texto para permitir la inclusión de atributos adicionales a los elementos, previa aprobación por parte del MOPTT. En 4.1 se agrega referencia a la sección 2.3.
- Se amplía el punto 4.1.1 para considerar Atributos que no forman parte del Elemento de Sistema, y se estandarizan cuatro configuraciones de atributos independientes de la aplicación.
- En las secciones 4.2 y 5.4, último párrafo, se agrega referencia a sección 2.3.
- En la sección 6.11, último párrafo, se agrega referencia a MOPTT-ST6.
- En la sección 7.1, párrafo final, se ajusta texto para hacer referencia al convenio de Televía.
- En 7.3.2 se ajusta el texto del último párrafo.
- En 7.3.3, se ajusta el texto para acomodarlo a las condiciones generales del convenio Televía.
- En las secciones 7.4, 7.4.1, 7.4.2 y 7.4.4 se agrega la posibilidad de usar esquemas alternativos de operación de las listas negra, gris y verde, luego de su aprobación por el MOPTT. En 7.3 se agrega referencia a la sección 2.3.
- En la sección 7.5, se modifica el texto para que en caso de discrepancia entre la clase declarada en el transponder y la medida por el sistema de clasificación, se aplique la tarifa de acuerdo a lo medido por el sistema de clasificación.
- Se agrega la sección 7.6, sobre los bits R y T en OBEConfiguration.
- Se agrega la sección 7.7, que especifica requerimiento al transponder para que en su carcasa lleve impreso el valor de Contract SerialNumber, en un formato definido.
- En el capítulo 8 y en el Anexo B, sección B.2, se eliminan las referencias al sitio web del MOPTT.
- En el Anexo A, secciones A.2.1, A.2.2.2 y A.2.3.2, y en los títulos de las figuras A.2.2 y A.2.3, se mejora el texto usando la terminología de transponder nativo y foráneo.
- En el Anexo A, sección A.2.3.2.2, párrafo 5 ítem 2, se corrige definición de contrato de duración indefinida.
- En el Anexo B, sección B.1 se agrega referencia a nuevo Anexo F, se amplía la descripción de los atributos OBEConfiguration y OBEGroupID, y se agregan los atributos ManufacturingSerialNumber / ManufacturerSerialNumber, 125 (Privado), ActivityTimer / ActivityTimerMOP, BatteryInsertionDate / BatteryInsertionDateMOP, NumberOf

Wake-ups, NumberOfReleases y NumberOfVSTs. En el dato ManufacturerId del atributo OBEConfiguration se agrega Telvent y se actualiza referencia al sitio del CEN TC278.

- En el Anexo B, secciones B.2 y B.3, se coloca bajo la responsabilidad del MOPTT la definición del valor de EFC-ContextMark, para evitar interpretaciones a que quedaba sujeto el texto en versiones anteriores de ST1.
- En B.2 se define valor para el atributo ContractSerialNumber y se define formato para su impresión. Además se agrega la clase 4 al atributo VehicleClass.
- En el Anexo B, sección B.4 se coloca bajo la responsabilidad del MOPTT la definición del valor de PM-ContextMark, para evitar interpretaciones a que quedaba sujeto el texto en versiones anteriores de ST1.
- En el Anexo B, sección B.5 se coloca bajo la responsabilidad del MOPTT la definición del valor de Private-ContextMark, para evitar interpretaciones a que quedaba sujeto el texto en versiones anteriores de ST1.
- En el Anexo C, se ajustan descripciones de bytes “fragmentation header”.
- En el Anexo C, sección C.2.1 se agrega comando opcional GET_NONCE.request
- En el Anexo C, secciones C.2.2 y C.2.3, se agrega respuesta GET_NONCE.response opcional.
- En el Anexo C, sección C.2.5, tercer servicio, GET.request opcional al elemento de sistema, se muestra el comando de lectura del atributo ActivityTimer, para las cuatro configuraciones de AIAs.
- En el Anexo C, secciones C.2.6 y C.2.7, tercer servicio, GET.response opcional al elemento de sistema, se muestra la respuesta con el atributo ActivityTimer, para las cuatro configuraciones de AIAs.
- En el Anexo C, sección C.2.8, se agrega SET.request opcional para bajar el Tamper bit, y se complementa descripción.
- En el Anexo C, secciones C.2.9 y C.2.10, se agrega SET.response opcional por bajada del Tamper Bit, y se complementa descripción.
- Se agrega el Anexo F, que define las 4 configuraciones de atributos independientes de la aplicación.

V 1.35 Enero 10, 2005 Séptima Versión. Cambios introducidos:

- En la sección 4.1, se permite la existencia de atributos aislados cuya lectura no requiere de credenciales de acceso.
- En las secciones 4.1.2, 4.1.3, 4.1.4 y 4.1.5, Tablas 4.2, 4.3, 4.4 y 4.5 respectivamente, se introducen cambios a las notas al pie de página.

- En la sección 7.6, la presencia del bit R es opcional.
- En la sección 7.7 se define que el formato del código de barras debe ser CODE 128.
- En Anexo A, sección A.2.2.1, se cambia la redacción del texto.
- En Anexo A, sección A.2.2.2, se ajusta la figura A.2.2 en la fase de Autenticación Fiscal y se colocan notas aclaratorias a continuación de dicha figura.
- En Anexo B, sección B.1, Atributo OBEGroupID, se cambia el texto para indicar valores de AttrID y las condiciones de acceso.
- En Anexo B, sección B.1, Atributo BatteryInsertionDate / BatteryInsertionDateMOP se corrige AttrID=16_{10}
- En Anexo B, sección B.2, Atributo ContractValidity, se corrige definición del dato elemental ContractExpiryDate y se entregan más detalles en los comentarios.
- En Anexo B, sección B.2, Atributo EquipmentStatus, se modifica el comentario del dato elemental GreenList
- En Anexo C, sección C.2.5, se agrega en el ejemplo de GET. Request opcional al Elemento de Sistema, el atributo BatteryInsertionDate.
- En Anexo C, sección C.2.6, se agrega en el ejemplo de GET. Response opcional al Elemento de Sistema, el atributo BatteryInsertionDate.
- En Anexo C, sección C.2.7, se agrega en el ejemplo de GET. Response opcional al Elemento de Sistema, el atributo BatteryInsertionDate.
- En Anexo C, sección C.2.8, para la configuración 4 de AIAs, se corrigen valores en bytes 85 y 97, el byte 90 se integra a la credencial de acceso y se modifica el comentario de los bytes 86 a 89.
- En Anexo C, sección C.2.9, para la configuración 4 de AIAs, se corrige valor en byte 22.
- En Anexo C, sección C.2.10, para la configuración 4 de AIAs, se corrige valor en byte 21.
- En Anexo F, se corrige numeración de secciones.
- En Anexo F, sección F.1, se agrega a la nota 2 al pie de la página: ROnAC, sólo lectura sin credenciales de acceso.
- En Anexo F, sección F.4, Tabla F.4: se reorganiza la tabla para indicar que TransponderSerialNumber y BatteryInsertionDate son datos del Atributo Privado 125. Se corrigen longitud de ActivityTimer: 6 bytes, AttrID de OBEGroupID: 37_{10}, accesos de OBEGroupID y Atributo Privado 125: RonAC.
- En Anexo F, sección F.4.1, se modifica texto para señalar que la lectura del Atributo Privado 125 es sin credenciales de acceso.

2 Introducción

2.1 GENERAL

Los modernos Sistemas de Cobro de tarifas, y otras aplicaciones dentro del ámbito conocido como Sistemas Inteligentes de Transporte o ITS, consideran equipamiento electrónico instalado en la carretera, que se comunica por radio con una pequeña unidad denominada “tag” o “transponder” montada en el parabrisas de cada vehículo.

Para su programa de concesiones, el Ministerio de Obras Públicas se ha fijado como objetivo establecer un ambiente en el cual la interoperabilidad entre concesiones sea posible, de manera que un mismo transponder pueda interactuar con los sistemas de cobro electrónicos y de otras aplicaciones ITS de cualquier concesión. Para ello, la comunicación radial entre los equipos de carretera y el transponder, y asimismo el intercambio de datos que se efectúa entre ellos, requieren de especificaciones detalladas y precisas. Es de conocimiento que aún cuando distintas instalaciones operen al amparo de un mismo estándar, pueden aparecer problemas de incompatibilidad, únicamente por una falta de regulación en todo el proceso de instauración de la tecnología escogida.

Debido a esto, y para continuar con un proceso que asegure la instauración de tecnología que sea claramente interoperable, el MOPTT ha impulsado la definición detallada de la norma a ser aplicada en el ámbito nacional.

2.2 OBJETIVOS

El presente documento tiene como objetivo fundamental establecer las condiciones de aplicación de los estándares relacionados con la comunicación de corto alcance *o DSRC*, entre los puntos de cobro y el transponder, en las obras concesionadas del Ministerio de Obras Públicas de Chile y en otras aplicaciones ITS.

Abarca temas relacionados con el cumplimiento con estándares de comunicación específicos, las funciones y comandos que deben ser soportados por el transponder, la estructuración de su memoria, los datos usados y los mecanismos de seguridad. Presenta la Transacción Nacional de Peaje Interoperable, y transacciones correspondientes a otras aplicaciones en el ámbito de ITS.

Adicionalmente y con el fin de afianzar la interoperabilidad, establece condiciones para la homologación de equipos bajo los requerimientos planteados en el presente documento.

2.3 NUEVAS VERSIONES Y DOCUMENTOS COMPLEMENTARIOS

Modificaciones a la presente especificación serán publicadas como una nueva versión de la misma. Además, el MOPTT emitirá documentos complementarios con asignaciones

que pueden variar en el tiempo pero que son necesarias para la interoperabilidad. Asimismo, cuando el MOPTT haya aprobado una solicitud de una o más concesiones para usar variantes conforme a lo señalado esta especificación, el MOPTT distribuirá a todas las concesiones un documento complementario con lo aprobado, indicando el plazo y condiciones para su instalación, y cubriendo particularmente todo lo que pueda afectar la interoperabilidad.

2.4 DEFINICIONES

Atributo Información formada por una secuencia de uno o más datos primarios.

Elemento Conjunto coherente de datos y funcionalidad, compuesto de Atributos. El direccionamiento de los Elementos de una aplicación se efectúa mediante identificadores de Elemento

Foráneo Designación que se aplica a un transponder cuando pasa por un RSE de una concesión diferente a la que lo emitió.

Nativo Designación que se aplica a un transponder cuando pasa por un RSE de la misma concesión que lo emitió.

Televía Transponder o dispositivo electrónico que cumpliendo los requisitos técnicos y operativos establecidos por el MOPTT, al estar instalado en un vehículo, permite su registro al pasar por un punto de cobro de una vía concesionada con peaje electrónico.

Transacción Intercambio total de información entre el equipamiento de carretera o RSE y el transponder del vehículo, necesario para completar una operación de cobro de peaje o de otra aplicación ITS a través del enlace DSRC.

2.5 ABREVIACIONES

En el texto se emplean las siguientes abreviaciones:

3-DES Algoritmo de cifrado de datos Triple-DES, descrito en el estándar ISO 11568, sección 4.2.

AIA Atributo independiente de la aplicación, residente en un transponder. Ejemplos: OBEConfiguration definido en [CEN – L7], Activity Timer o contador de tiempo que el transponder ha estado activo.

AID “Application Identifier”, número con que se identifican las diferentes aplicaciones según [CEN – L7]. Ejemplos: 1 = cobro electrónico de tarifas, 2 = gestión de flotas, 6 = gestión de estacionamientos, etc

BST “Beacon Service Table”, tabla de servicio de los equipos fijos en la carretera o RSE. Consiste en una secuencia de datos transmitida por el RSE para indicar los servicios que ofrece.

CEN Comité Europeo de Normalización

DES "Data Encryption Standard", estándar de cifrado de datos, de acuerdo a la norma ANSI X3.92.

DSRC "Dedicated Short Range Communication", comunicaciones dedicadas de corto alcance. Son las comunicaciones entre los equipos de carretera RSE y el equipo a bordo del vehículo OBE.

EFC "Electronic Fee Collection", cobro electrónico de tarifas. Esta es la denominación que se aplica al peaje electrónico en la terminología europea.

EID "Element Identifier", número entre 0 y 127 con que se identifican los diferentes Elementos en el interior del transponder. El Elemento de sistema se identifica con el valor 0.

ITS "Intelligent Transportation Systems", Sistemas Inteligentes de Transporte. Comprende un rango amplio de tecnologías, entre ellas electrónica, comunicaciones, control, procesamiento de información, aplicadas en forma coherente a los sistemas de transporte.

MAC "Message Authentication Code", valor criptográfico calculado sobre un conjunto de datos. El valor de MAC puede ser considerado como una firma, que permite garantizar la integridad de los datos.

MMI "Man Machine Interface", interfaz entre el equipo y el usuario. En un transponder puede ser un display alfanumérico, indicadores luminosos o acústicos, etc.

MOPTT Ministerio de Obras Públicas, Transportes y Telecomunicaciones de la República de Chile

OBE "On Board Equipment", equipo a bordo del vehículo. Constituido típicamente por el transponder.

RSE "Road Side Equipment", equipos fijos en la carretera. Los equipos dedicados a la comunicación muchas veces se designan con el término "beacon".

SAM "Secure Application Module", módulo de aplicación seguro. Módulo de procesamiento electrónico sellado e inviolable, usado para proteger información crítica como claves de seguridad.

TC278 Comité Técnico 278 del CEN

VST "Vehicle Service Table", tabla de servicio del vehículo. Consiste en una secuencia de datos transmitida por el OBE para indicar los servicios que puede atender.

3 Definición General para el Cobro Electrónico

3.1 DOCUMENTOS DE REFERENCIA

El presente documento incorpora provisiones de determinados estándares y asimismo de otras publicaciones. Estos documentos son citados en los lugares apropiados del texto, presentándose a continuación la nómina de ellos. En caso de modificaciones o revisiones de cualquiera de estas publicaciones, se aplicarán al presente documento las últimas ediciones de ellas. En el caso de existir inconsistencias entre los documentos, prima lo establecido en los estándares sobre las especificaciones complementarias de la industria.

[CEN – L1] **prEN 12253: 2002**
Road Traffic and Transport Telematics (RTTT)
Dedicated Short Range Communication (DSRC)
DSRC Physical Layer using Microwave 5.8 GHz

[CEN – L2] **prEN 12795: 2002**
Road Traffic and Transport Telematics (RTTT)
Dedicated Short Range Communication (DSRC)
DSRC Data Link Layer: Medium Access and Logical Link Control

[CEN – L7] **prEN 12834: 2002**
Road Traffic and Transport Telematics (RTTT)
Dedicated Short Range Communication (DSRC)
Application Layer

[CEN – PR] **Draft prEN 13372: 2002**
Road Traffic and Transport Telematics (RTTT)
Dedicated Short Range Communication (DSRC)
DSRC Profiles for RTTT Applications

[ISO – EFC] **Draft prEN ISO 14906: 2002**
Road Traffic and Transport Telematics (RTTT)
Electronic Fee Collection (EFC)
Application Interface Definition for Dedicated Short Range Communications

[ISO – NDS] **ENV ISO 14816: 1997**
Road Traffic and Transport Telematics (RTTT)
AVI/AEI: Numbering and Data Structures

[ISO – CC] **ISO 3166-1: 1987 (E)**
Codes for the Representation of Names of Countries and their Subdivisions.
Part 1: Country Codes

[ISO – MAC] **ISO 8731-1: 1987 (E)**
Banking
Approved Algorithms for Message Authentication

[GSS – 2.0] **GSS: Feb. 1999**
Global Specification for Short Range Communication
Bosch Telecom GmbH, Alcatel CGA Transport, Combitech Traffic Systems AB

[A1] **TR 4001 A1: June 12, 1999**
Version ER9_1.3
Interoperable EFC Transaction Using Central Account Based on DSRC
Alcatel, Combitech, Kapsch, CSSI

[CARDME – 3] **TR 4102: May 8, 2000**
D3.3
Specification of an Interoperable European EFC Service

[MOPTT – ST1-1] **MOPTT ST1-1: Agosto 4, 2004**
Sistemas Electrónicos de Cobro y Otras Aplicaciones
Context Marks en Servicio en Concesiones MOPTT
Ministerio de Obras Públicas, Transportes y Telecomunicaciones, CHILE

[MOPTT – ST2] **MOPTT ST2: July 15, 2002**
Electronic Fee Collection and Other Applications
Conformance Tests to the Specification for Interoperability in the Beacon - Transponder Transaction
Public Works, Transport and Telecommunications Ministry, CHILE

[MOPTT – ST3] **MOPTT ST3: Junio 3, 2004**
Sistemas Electrónicos de Cobro y Otras Aplicaciones
Gestión de las Claves de Seguridad
Ministerio de Obras Públicas, Transportes y Telecomunicaciones, CHILE

[MOPTT – ST6] **MOPTT ST6: Agosto 10, 2004**
Sistemas Electrónicos de Cobro y Otras Aplicaciones
Generación de Claves Maestras, Interfaz de Exportación de Claves
Ministerio de Obras Públicas, Transportes y Telecomunicaciones, CHILE

3.2 COMUNICACIONES DSRC

Mediante comunicaciones radiales de corto alcance, conocidas también como "DSRC", se llevan a cabo transferencias bidireccionales de datos, entre el equipamiento instalado en el punto de cobro (RSE) y el tag o transponder (OBE).

Para alcanzar la interoperabilidad entre equipos provistos por fabricantes diferentes, en los sistemas telepeaje de obras concesionadas por el MOPTT, las comunicaciones DSRC se regirán según lo establecido por el Comité Técnico TC278 del CEN en la siguiente documentación:

Nivel 1 DSRC o Físico: [CEN – L1]

Nivel 2 DSRC o del Data Link: [CEN – L2]

Nivel 7 DSRC o de la Aplicación: [CEN – L7]

Perfiles DSRC: [CEN – PR]

Las variantes de implementación que son posibles bajo los estándares anteriores, permiten llegar a sistemas que en último término no son compatibles. Para subsanar el problema, es necesario acotar el espectro de opciones permitidas, mediante la adición de especificaciones complementarias, las que se definen en el Anexo E, sección E.1.

3.3 APLICACIÓN DE COBRO ELECTRÓNICO DE PEAJE

Para posibilitar la interoperabilidad, la aplicación de cobro electrónico de peaje o EFC requiere de una interfaz estandarizada con las comunicaciones DSRC.

Para este propósito, los sistemas de telepeaje de las obras concesionadas por el MOPTT y basados en comunicaciones DSRC, deberán emplear una interfaz de acuerdo con lo prescrito en:

Interfaz EFC: [ISO – EFC]

Numeración y Estructuras de Datos: [ISO – NDS]

[ISO – EFC] ofrece múltiples posibilidades, que pueden conducir a soluciones no interoperables. Con el fin de definir en forma precisa cuáles son las opciones que en definitiva se utilizan, y adicionalmente para adoptar un mecanismo de seguridad uniforme, será obligatorio ceñirse a lo establecido en el Anexo E, sección E.2.

Inicialmente, la transacción de cobro de peaje será para cuenta central de clientes, sin perjuicio que en el futuro puedan usarse otras modalidades de cuentas de clientes, pero sólo una vez que se haya acordado con el MOPTT la respectiva especificación de la Transacción.

3.3.1 TRANSACCIÓN CARDME – 3

El marco definido en la sección E.2.1 del Anexo E ha sido aplicado con éxito a la generación de la Transacción especificada en [CARDME – 3]. Ella responde a los requerimientos identificados para un servicio de cobro de peaje interoperable a través de Europa, en un ambiente multinacional y multioperador.

La solución definida en [CARDME – 3] está siendo promovida en los diferentes países de la Comunidad Europea, con el objetivo de alcanzar interoperabilidad en los sistemas de telepeaje. La validez de la Transacción propuesta ha quedado confirmada a través de pruebas en transponders diseñados de acuerdo con [A1].

3.3.2 TRANSACCIÓN NACIONAL DE PEAJE INTEROPERABLE PARA LAS CONCESIONES DEL MOPTT

El mismo marco presentado en E.2.1 del Anexo E, fija los límites en los que se sitúa la Transacción Nacional de Peaje Interoperable para las concesiones MOPTT, que se define en el Anexo A. En esencia, es una Transacción similar a la propuesta por [CARDME – 3], con algunas variaciones para adaptarla a las condiciones locales.

Comparada con la Transacción de [CARDME – 3], la Transacción del MOPTT agrega una fase de autenticación fiscal, necesaria para resguardar los intereses del Estado, pero en cambio simplifica el tratamiento de los vehículos de carga. Solamente emplea funciones dentro de las establecidas en [A1], puesto que otras funciones no pueden ser ejecutadas por los transponders [A1] disponibles en el mercado. En cuanto a los Atributos empleados, de preferencia son los especificados en [A1], pero se incluyen otros, algunos de tipo privado según lo establecido en [ISO – EFC].

3.4 OTRAS APLICACIONES

Además de la Transacción Nacional de Peaje Interoperable, el transponder para las concesiones del MOPTT deberá ser capaz de ejecutar las transacciones adicionales definidas en el Anexo A.

4 Datos en el Transponder

4.1 ELEMENTOS

La memoria del transponder deberá organizarse sobre la base de los Elementos especificados en la Tabla 4.1. No podrán usarse Elementos adicionales a los indicados, sin una autorización previa por escrito emitida por el MOPTT. Si una o más concesiones desean agregar Elementos nuevos, o colocar Atributos nuevos en los Elementos, deberán elevar ante el MOPTT una solicitud donde se detalle la finalidad, así como el contenido del o de los Elementos y/o Atributos propuestos. Teniendo como objetivo mantener el ambiente de interoperabilidad, el MOPTT estudiará las respectivas solicitudes, podrá convenir eventuales modificaciones, y comunicará su decisión a las partes interesadas. En todo caso, el MOPTT se reserva el derecho a rechazar cualquier solicitud sin expresión de causa. Si la solicitud es aprobada, se procederá según lo indicado en 2.3.

TABLA 4.1 Elementos en el Transponder MOPTT

Elemento	*ApplicationContext Mark*	*EID*	*Identificación de la Aplicación (AID)*	*Protección de Acceso*
Elemento de Sistema	(No aplicable)	0	0: Indep. de la Aplicac.	Clave DES
Cobro de Peaje Interoperable	EFC-ContextMark	n1	1: EFC	Clave DES
Del Emisor del Transponder	EFC-ContextMark	n2	1: EFC	Clave DES
Gestión de Estacionamientos	PM-ContextMark	n3	6:parking-management	Clave DES
Sonda de Tráfico	Private-ContextMark	n4	29: private	Clave DES

Los Atributos ApplicationContextMark quedan asociados con los respectivos identificadores de Elemento EID, en la lista de aplicaciones que el transponder transmite en la VST. La protección de acceso indicada en la Tabla 4.1 se refiere a las condiciones de acceso de los Atributos que integran el Elemento, durante el desarrollo de la Transacción. Todos los Elementos deberán estar protegidos, de manera que el procesamiento de cualquier comando dirigido a un Elemento, requiera de credenciales de acceso según se define en la especificación [A1]. Sujeto a la aprobación del MOPTT, podrán existir atributos aislados cuya lectura no requiera de credenciales de acceso.

4.1.1 ATRIBUTOS INDEPENDIENTES DE LA APLICACIÓN Y ELEMENTO DE SISTEMA

Los Atributos independientes de la aplicación pueden formar parte del Elemento de Sistema, o pueden estar ubicados en un espacio alternativo, según lo definido para cada modelo de transponder por el respectivo fabricante. Ejemplos de estos Atributos son OBEConfiguration, OBEGroupID, contador del tiempo activo del transponder, etc. Por la eventual influencia de estos Atributos y/o del Elemento de Sistema en la funcionalidad del transponder, el MOPTT ha estandarizado cuatro configuraciones de ellos, las que se detallan en el Anexo F. Los modelos de transponders que sean sometidos al proceso de

homologación en conformidad con lo estipulado en el capítulo 9 de la presente especificación, deberán usar obligatoriamente una de dichas cuatro configuraciones.

Los equipos RSE deberán ser capaces de usar la funcionalidad completa que se deriva de las cuatro configuraciones de AIAs del Anexo F, con los transponders operando tanto en la modalidad de nativo como foráneo. El RSE deberá ser configurable mediante parámetros, con los que el operador pueda habilitar o deshabilitar el uso de cualquier funcionalidad producto de las configuraciones del Anexo F.

4.1.2 ELEMENTO DE COBRO DE PEAJE INTEROPERABLE

Los Atributos incluidos en este Elemento se presentan en la Tabla 4.2. Todos lo Atributos señalados deberán estar presentes, y sólo podrán existir Atributos adicionales previa aprobación por parte del MOPTT.

TABLA 4.2 Elemento de Cobro de Peaje Interoperable (Aplicación 1)				
Nombre del Atributo	***AttrID (1)***	***Longitud (Bytes)***	***Acceso (3)***	***Comentarios***
EFC-ContextMark	0_{10}	6	(2)	Emitido en la VST
ContractSerialNumber	1_{10}	4	RO	
ContractValidity	2_{10}	6	RO	
ReceiptServicePart	5_{10}	13	R/W	
SessionClass	6_{10}	2	R/W	
ReceiptAuthenticator	13_{10}	5	R/W	
VehicleClass	17_{10}	1	RO	
EquipmentStatus	26_{10}	2	R/W	
Spare	98_{10}	13	R/W	Espacio de Reserva
ElementAuthenticationKeyA1	111_{10}	8 (4)	NA	Claves de entidad emisora del transponder
ElementAuthenticationKeyA2	112_{10}	8 (4)	NA	
ElementAuthenticationKeyF1	113_{10}	8 (4)	NA	Claves del MOPTT para autenticación fiscal
ElementAuthenticationKeyF2	114_{10}	8 (4)	NA	
ElementAuthenticationKeyI1	115_{10}	8 (4)	NA	Claves para interoperabilidad
ElementAuthenticationKeyI2	116_{10}	8 (4)	NA	
ElementAuthenticationKeyI3	117_{10}	8 (4)	NA	
ElementAuthenticationKeyI4	118_{10}	8 (4)	NA	
ElementAccessKey	120_{10}	8 (4)	NA	

[1] AttrID: Número identificador del Atributo [2] Depende del modelo de tag, puede ser NDA o ROnAC
[3] NDA: Sin acceso directo RO: Sólo lectura ROnAC: Sólo lectura, sin credencial de acceso
R/W: Lectura y escritura NA: Sin acceso
[4] Los algoritmos de seguridad en transponders según [A1] son de tipo DES y por lo tanto las claves respectivas son de 8 bytes. En el caso de que el transponder emplee algoritmos 3-DES, las claves serán de 16 bytes de longitud; para mantener compatibilidad con la especificación [A1], las claves deberán definirse con las mitades izquierda y derecha iguales.

4.1.3 ELEMENTO DEL EMISOR DEL TRANSPONDER

Los transponders empleados en las concesiones del MOPTT deberán contener un Elemento reservado al emisor del transponder, presentado en la Tabla 4.3. Todos los Atributos señalados deberán estar presentes, y sólo podrán existir Atributos adicionales previa aprobación por parte del MOPTT. El uso de este Elemento queda sujeto a lo dispuesto en 4.1.

TABLA 4.3 Elemento del Emisor del Transponder (Aplicación 1)				
Nombre del Atributo	**AttrID (1)**	**Longitud (Bytes)**	**Acceso (3)**	**Comentarios**
EFC-ContextMark	0_{10}	6	(2)	Emitido en la VST
Scratchpad	96_{10}	6	R/W	
ElementAccessKey	120_{10}	8 (4)	NA	

Ver notas (1) a (4) al pie de la página 19.

4.1.4 ELEMENTO PARA GESTIÓN DE ESTACIONAMIENTOS

Los transponders empleados en las concesiones del MOPTT deberán contener un Elemento dedicado a una gestión de estacionamientos de tipo simple. Los Atributos incluidos en este Elemento se presentan en la Tabla 4.4. Todos lo Atributos señalados deberán estar presentes, y sólo podrán existir Atributos adicionales previa aprobación por parte del MOPTT.

TABLA 4.4 Elemento para Gestión de Estacionamientos (Aplicación 6)				
Nombre del Atributo	**AttrID (1)**	**Longitud (Bytes)**	**Acceso (3)**	**Comentarios**
PM-ContextMark	0_{10}	6	(2)	Emitido en la VST
ContractSerialNumber	1_{10}	4	RO	
ElementAuthenticationKey	111_{10}	8 (4)	NA	
ElementAccessKey	120_{10}	8 (4)	NA	

Ver notas (1) a (4) al pie de la página 19.

4.1.5 ELEMENTO PARA SONDA DE TRÁFICO

Los transponders empleados en las concesiones del MOPTT deberán contener un Elemento para sonda de tráfico. Los Atributos incluidos en este Elemento se presentan en la Tabla 4.5. Todos lo Atributos señalados deberán estar presentes, y sólo podrán existir Atributos adicionales previa aprobación por parte del MOPTT.

TABLA 4.5 Elemento para Sonda de Tráfico (Aplicación 29)				
Nombre del Atributo	***AttrID (1)***	***Longitud (Bytes)***	***Acceso (3)***	***Comentarios***
Private-ContextMark	0_{10}	6	(2)	Emitido en la VST
TemporaryID	97_{10}	3	R/W	
ElementAccessKey	120_{10}	8 (4)	NA	

Ver notas (1) a (4) al pie de la página 19.

4.2 VALORES DE LOS ATRIBUTOS

Los valores asignados a los Atributos, y a los datos individuales que los componen, se detallan en el Anexo B. Dependiendo del caso, la asignación responde a una de las siguientes tres alternativas:

- El valor se encuentra especificado en las referencias identificadas en la Sección 3.1
- El valor ha sido definido por el MOPTT
- El valor está abierto para que sea generado directamente por la concesión emisora del transponder.

La última alternativa no podrá ser usada para cambiar valores que ya se encuentren definidos en el Anexo B. Todas las asignaciones que sean elaboradas por una concesión, deberán ser registradas en el MOPTT a lo menos seis meses antes del inicio de la distribución de los transponders que las empleen. El MOPTT llevará un catálogo de los valores que las diferentes concesiones hayan asignado y los publicará en documentos complementarios, según se define en 2.3.

5 Transacciones

Cualquiera que sea la aplicación a considerar con los transponders para las concesiones del MOPTT, las transacciones se componen de diferentes fases que se ejecutan secuencialmente:

- Inicialización
- Núcleo de la Transacción
- Término de la Transacción

5.1 INICIALIZACIÓN

Durante la inicialización se da comienzo a la sesión de comunicaciones y se declaran las aplicaciones soportadas por el punto de cobro o RSE y por el transponder.

El punto de cobro emite en forma periódica una señal denominada BST, que contiene los respectivos códigos de identificación AID de las aplicaciones residentes en él. Cuando un transponder OBE ingresa a la zona de comunicaciones de la antena, analiza la BST y determina si está en condiciones de atender alguna de las aplicaciones presentadas en ella. Cuando ello es posible, el transponder solicita mediante el mensaje PrWRq, que se le asigne una ventana de tiempo para iniciar la Transacción. RSE define entonces la ventana a través del mensaje PrWA.

La respuesta del transponder, denominada VST, lleva una lista de todas las aplicaciones que estando incluidas en la BST, también pueden ser atendidas por el transponder. Cada componente de la lista contiene el identificador de la aplicación AID, seguido del número identificador del Elemento EID en que residen los Atributos de esa aplicación, y de un parámetro que contiene los Atributos ApplicationContextMark y otros dependientes de la aplicación.

5.2 NÚCLEO DE LA TRANSACCIÓN

El núcleo de la Transacción abarca fases que son específicas para cada aplicación, en las que se efectúa el intercambio de datos entre el RSE y el OBE. Una misma aplicación puede tener diferentes transacciones, dependientes de la funcionalidad implementada en cada una de ellas, y/o del tipo de transponder empleado.

5.3 TÉRMINO DE LA TRANSACCIÓN

El término de la Transacción comprende las acciones necesarias para finalizar la sesión de comunicaciones. Dependiendo de las condiciones de operación de aplicaciones determinadas, puede incluir fases específicas como por ejemplo el seguimiento de la posición del transponder en autopistas de flujo libre.

5.4 TRANSACCIONES RECONOCIDAS

El Anexo A presenta las transacciones reconocidas para los transponders de las concesiones del MOPTT. A la fecha de su emisión, cada transponder deberá ser capaz de ejecutar todas las transacciones que se encuentren reconocidas en la versión vigente del presente documento.

En el caso de que una o más concesiones deseen modificar una transacción reconocida o agregar otras nuevas, deberán elevar ante el MOPTT una solicitud donde se detalle la finalidad, así como la correspondiente codificación de la misma. Teniendo como objetivo mantener el ambiente de interoperabilidad, el MOPTT estudiará las respectivas solicitudes, podrá convenir eventuales modificaciones, y comunicará su decisión a las partes interesadas. En todo caso, el MOPTT se reserva el derecho a rechazar cualquier solicitud sin expresión de causa. Si la solicitud es aprobada, se procederá según lo indicado en 2.3.

5.5 CODIFICACIÓN DE LAS TRANSACCIONES

La codificación de las tramas correspondientes a las diferentes fases de las transacciones reconocidas se presenta en el Anexo C.

6 Esquema de Seguridad

De acuerdo con lo establecido en la especificación [A1], los datos se transmiten a través del canal DSRC en texto claro, sin cifrado. La seguridad se basa en el empleo de una credencial de acceso y de valores de autenticación MAC, calculados mediante algoritmos especificados en [ISO – MAC].

La Transacción emplea seguridad en ambas direcciones:

- Para leer y/o modificar los Atributos de un Elemento protegido en el transponder, el RSE debe presentar una credencial de acceso válida AC_CR al transponder.
- Para que los datos provenientes del transponder sean aceptados por el RSE, el transponder debe presentar un autenticador válido. Todas las transacciones utilizan autenticación doble, como se describe en el Anexo A, sección A.2.3.2.

En los puntos siguientes se presentan los aspectos más importantes del esquema de seguridad. Los detalles específicos de los procedimientos criptográficos se encuentran en [A1].

6.1 GRUPOS DE CLAVES DE SEGURIDAD

En el esquema de seguridad intervienen claves que sólo están en conocimiento de las entidades autorizadas para acceder a la memoria del transponder, así como para generar transacciones válidas.

Las claves maestras para efectuar estas operaciones residen en el RSE y son del tipo 3-DES. En cambio, en el transponder se usan claves DES, derivadas de las anteriores. Este esquema resguarda las claves maestras, que en ningún caso quedan disponibles en equipos entregados al público. Aún suponiendo que terceros lograran extraer de un transponder las claves DES, no ponen en jaque la seguridad del sistema, porque con los escasos datos recuperados, no es posible determinar los valores de las claves maestras.

Se distinguen cinco grupos de claves maestras:

- **Claves Maestras de Acceso a los Elementos MEAcK:** a excepción de la clave utilizada para el acceso al Elemento del emisor del transponder, ellas pertenecen al dominio interoperable, y son conocidas por el MOPTT y por todas las concesiones. En el transponder residen las claves derivadas ElementAccessKey o **EAcK**.
- **Claves Maestras de Autenticación del Emisor del Transponder MEAuKA:** sólo son conocidas por la concesión emisora del transponder, y no deberán ser comunicadas a terceras partes. En el transponder residen las claves derivadas ElementAuthenticationKeyA o **EAuKA**.
- **Claves Maestras de Autenticación del MOPTT MEAuKF:** sólo son conocidas por el MOPTT. En el transponder residen las claves derivadas ElementAuthenticationKeyF o **EAuKF**.

- **Claves Maestras de Autenticación Interoperables MEAuKI:** son conocidas por el MOPTT y por todas las concesiones (dominio interoperable). En el transponder residen las claves derivadas ElementAuthenticationKeyI o **EAuKI**.
- **Claves Maestras de Autenticación del Recibo grabado en el Transponder MReAuK:** sólo son conocidas por la concesión que ejecuta la transacción. En este caso, las claves derivadas **DeReAuK** no residen en el transponder, sino en el RSE.

6.2 ACCESO A UN ELEMENTO DE LA MEMORIA DEL TRANSPONDER

La figura 6.1 presenta el procedimiento con el que se calcula la credencial de acceso AC_CR, separadamente tanto en el RSE como en el transponder. Cuando el resultado coincide en ambos lados, el transponder permite el acceso al Elemento seleccionado.

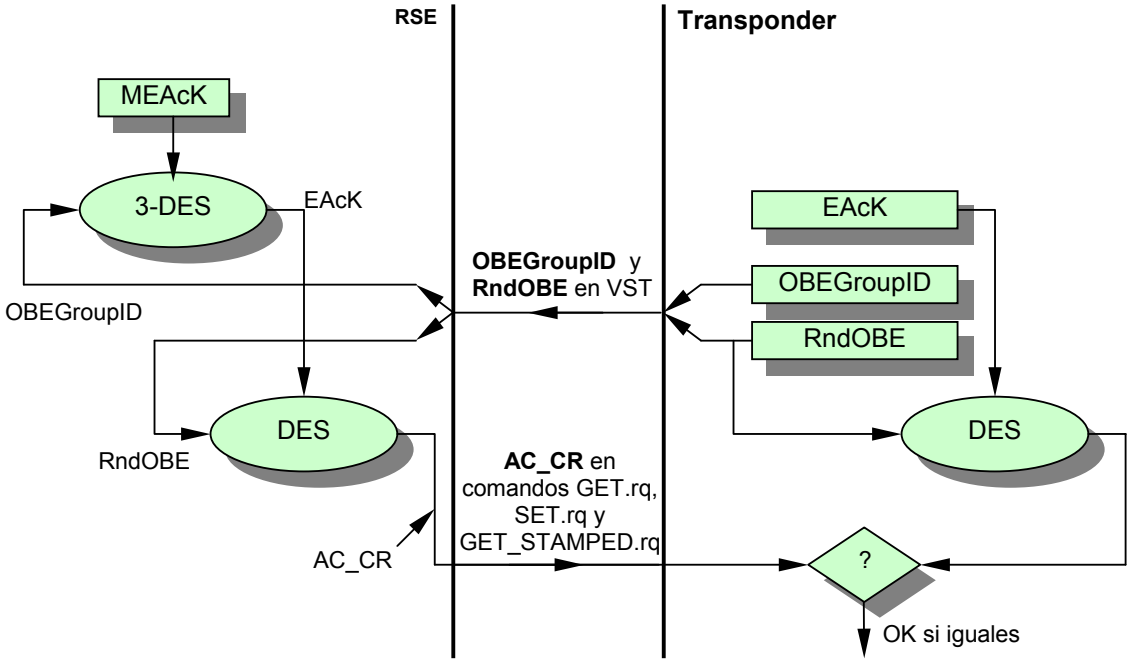

Figura 6.1 Acceso a la Memoria del Transponder

La clave de acceso EAcK residente en el transponder es derivada a partir de la clave maestra MEAcK y del valor del Atributo OBEGroupID. Por lo tanto, EAcK resulta igual en todos los transponders que tienen el mismo valor de OBEGroupID. Si el EAcK de un transponder llega a ser conocido, sólo queda comprometida la seguridad del grupo, en lo referente al acceso a los datos del Elemento. Para el sistema de concesiones del MOPTT, se especifican grupos que abarcan aproximadamente 500 transponders. Grupos de tamaño menor mejorarían este aspecto de la seguridad, pero degradan la privacidad de los usuarios.

Durante la Transacción, el transponder obtiene el valor dinámico de AC_CR mediante el algoritmo DES y el número aleatorio RndOBE, usando la clave interna EAcK. El cálculo es más complejo en el RSE, debido a que primero se determina el valor EAcK que corresponde al transponder, usando MEAcK y OBEGroupID. A continuación se calcula AC_CR aplicando el mismo procedimiento utilizado en el transponder.

6.3 AUTENTICACIÓN DEL TRANSPONDER

El procedimiento para determinar si un transponder es válido consiste en calcular el número OBEAuthenticator separadamente, tanto en el RSE como en el transponder, de acuerdo al procedimiento que se presenta en la figura 6.2. Cuando el resultado coincide en ambos lados, el transponder es considerado genuino.

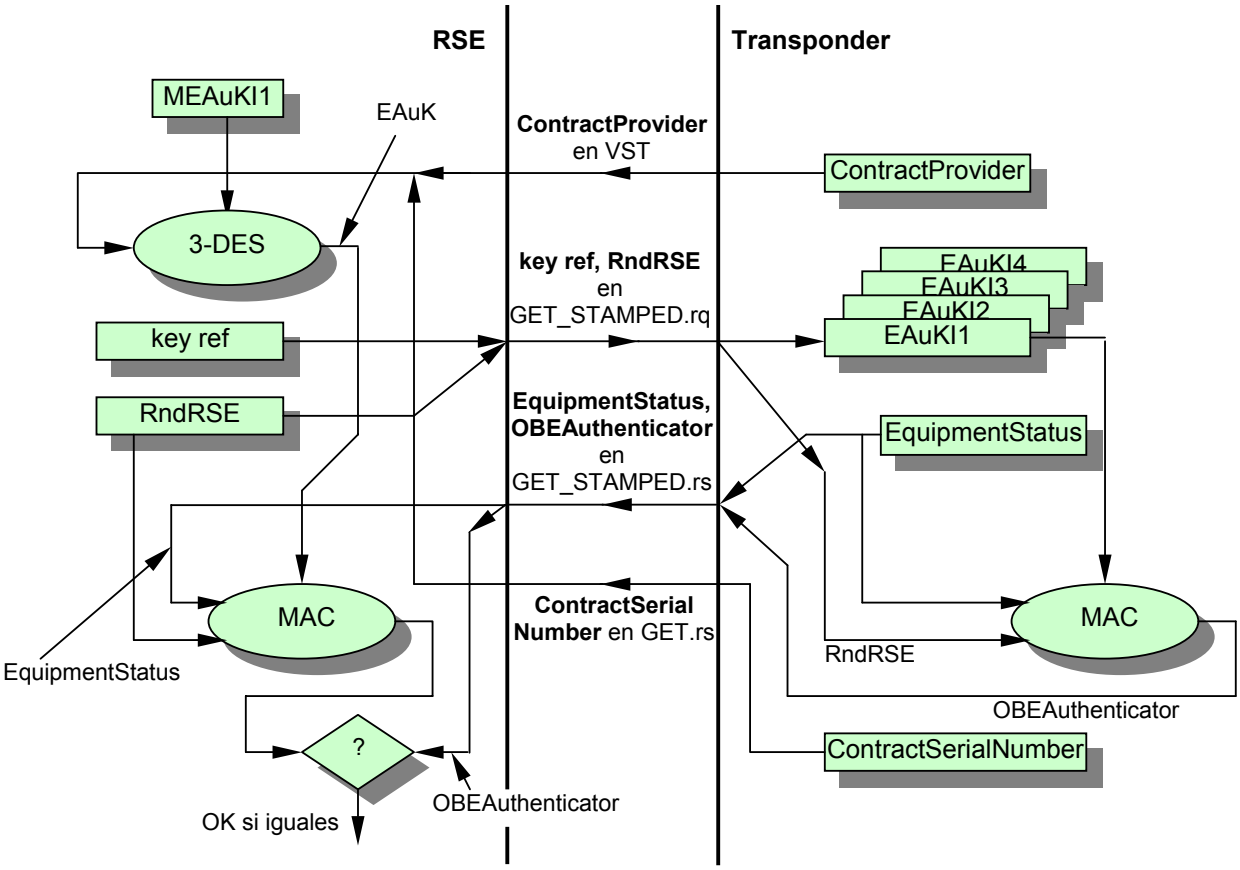

Figura 6.2 Autenticación del Transponder

Las claves de autenticación EAuKI residentes en el transponder son derivadas a partir de sendas claves maestras MEAuKI, con los valores de los Atributos ContractSerialNumber y ContractProvider. Con ello, las claves EAuKI son diferentes para cada transponder, de manera que si una de ellas llega a ser conocida, sólo queda comprometida la seguridad de ese transponder, en lo referente a la autenticación con esa clave.

Durante la Transacción, el transponder calcula el MAC dinámico OBEAuthenticator con el Atributo EquipmentStatus y el número aleatorio RndRSE, usando una de las claves internas EAuKI. En el RSE, primero se determina el valor EAuKI que corresponde al transponder, usando MEAuKI y los Atributos ContractSerialNumber y ContractProvider que envía el transponder. A continuación se calcula OBEAuthenticator siguiendo el mismo procedimiento usado en el transponder.

Todas las concesiones pueden verificar el valor de OBEAuthenticator, gracias a que las claves empleadas son del dominio interoperable. Se ha previsto que cuatro generaciones de claves EauKI residan en el transponder, las que son seleccionadas mediante la variable *key ref* a través de los valores 115 a 118. Ello permite cambiar la clave en uso si existen sospechas de que ha sido quebrada la que se encuentra en servicio.

Como se describe en el Anexo A, el emisor del transponder puede usar para el cálculo de OBEAuthenticator, claves del grupo EAuKA y MEAuKA, seleccionadas mediante *key ref* igual a 111 o 112.

6.4 AUTENTICACIÓN FISCAL

El procedimiento para generar el autenticador fiscal FiscalAuthenticator es idéntico al presentado en 6.3 en relación con OBEAuthenticator, a excepción de lo siguiente:

- En lugar de las claves EAuKI y MEAuKI, se utilizan las claves del dominio del MOPTT EAuKF y MEAuKF.
- El transponder incorpora en este caso dos generaciones de claves EauKF, seleccionadas con *key ref* igual a 113 o 114. El resultado del cálculo en el transponder es FiscalAuthenticator en lugar de OBEAuthenticator.
- La revisión del valor de FiscalAuthenticator no se realiza en el RSE, sino en el sistema del MOPTT, que efectúa estas operaciones fuera de línea y discrecionalmente.

6.5 AUTENTICACIÓN DEL CONTRATO

El procedimiento para obtener el autenticador del contrato ContractAuthenticator es idéntico al presentado en 6.3 en relación con OBEAuthenticator, exceptuado lo siguiente:

- En lugar de las claves EAuKI y MEAuKI, se utilizan las claves del dominio privado de la concesión emisora del transponder EAuKA y MEAuKA.
- El transponder incorpora en este caso dos generaciones de claves EauKA, seleccionadas con *key ref* igual a 111 o 112. El resultado del cálculo en el transponder es ContractAuthenticator en lugar de OBEAuthenticator.
- La revisión del valor de ContractAuthenticator no se realiza en el RSE, sino en el sistema de la concesión emisora del transponder, que efectúa estas operaciones fuera de línea.

6.6 AUTENTICACIÓN DEL RECIBO

El procedimiento para determinar el valor de ReceiptAuthenticator se presenta en la figura 6.3. El transponder se limita a tomar nota y grabar el valor, pero no lo puede comprobar, al no conocer la clave con que fue determinado.

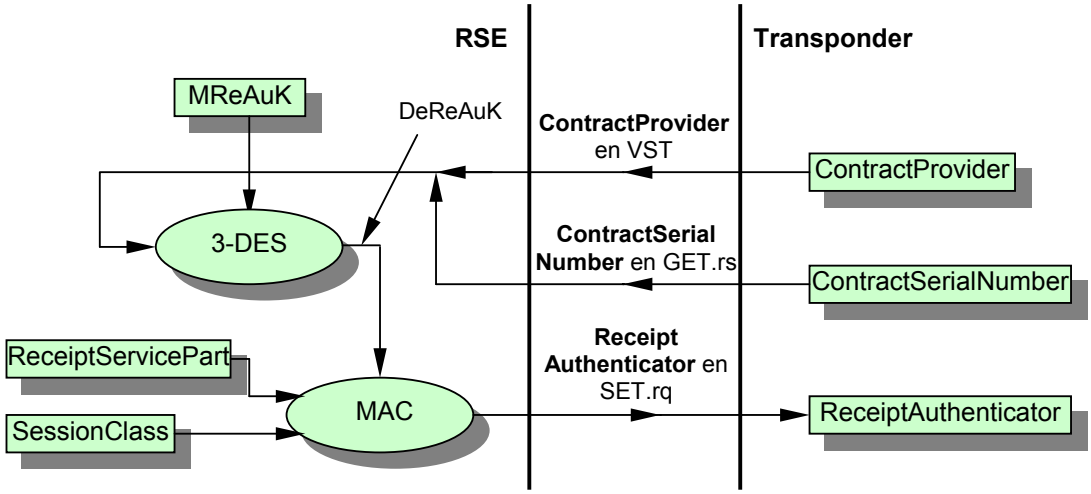

Figura 6.3 Autenticación del Recibo.

Durante la Transacción, el RSE determina primero el valor DeReAuK que corresponde al transponder, usando MReAuK y los Atributos ContractSerialNumber y ContractProvider que envía el transponder. A continuación calcula el MAC ReceiptAuthenticator con los Atributos ReceiptServicePart y SessionClass, usando la clave derivada DeReAuK. Sólo quien conoce MReAuK puede recalcular o comprobar el valor de ReceiptAuthenticator.

6.7 VALORES DE RNDOBE Y RNDRSE

Los números aleatorios RndOBE y RndRSE introducen una variación dinámica a los valores de AC_CR, OBEAuthenticator, FiscalAuthenticator y ContractAuthenticator, la que impide el “replay” de casos previos.

Típicamente, RndOBE es generado en el transponder mediante un algoritmo que entrega una larga secuencia de valores pseudoaleatorios. Será obligatorio que esta secuencia sea inicializada en fábrica con un valor aleatorio, para que los diferentes transponders distribuidos al público operen a partir de puntos diferentes de la secuencia.

RndRSE deberá hacerse igual a la variable Time definida en [CEN – L7], valor de 32 bits que representa los segundos transcurridos desde el 1º de enero de 1970, 00:00 (UTC). Estrictamente, esto no corresponde a un número aleatorio, pero cumple con la función definida para RndRSE porque el “replay” de casos previos no es posible, gracias a que

Time nunca repite valores anteriores. Por otra parte, como RSE no conoce de antemano el momento en que se inicia una nueva Transacción, no puede repetir el valor de RndRSE de una Transacción anterior ejecutada con el mismo transponder, en la fase de autenticación de contrato.

6.8 CONTADOR DE TRANSACCIONES DEL TRANSPONDER

Como una medida adicional de seguridad, el Atributo EquipmentStatus deberá llevar en los 12 bits bajos un contador de las transacciones ejecutadas por el transponder. Este contador puede contribuir a probar determinadas situaciones de fraude, e identificar instancias en las que se ha quebrado la seguridad.

Durante la Transacción, el contador de Transacciones deberá ser incrementado preferentemente por el transponder mismo, o alternativamente por el RSE. El procedimiento siguiente permite la coexistencia de transponders con y sin la capacidad de efectuar dicho incremento, sin cambios en el RSE:

- Acciones en el RSE:
  - ✓ Se lee el valor de EquipmentStatus mediante un comando GET_STAMPED, durante la fase de Presentación.
  - ✓ El contador en los 12 bits bajos es incrementado, y se actualizan los flags en los 4 bits altos.
  - ✓ Se escribe en el transponder el valor actualizado de EquipmentStatus mediante un comando SET, durante la fase de emisión del Recibo.

- Acciones en el transponder:
  - ✓ En un transponder sin contador interno de transacciones:
    - Se actualiza el valor de EquipmentStatus del transponder con el valor de EquipmentStatus contenido en el comando SET de la fase de emisión del Recibo.
  - ✓ En un transponder dotado de contador interno de transacciones, éste es activado por el comando SET aplicado a EquipmentStatus:
    - Se descartan los 12 bits bajos del valor de EquipmentStatus contenido en el comando SET de la fase de emisión del Recibo.
    - Se actualizan los 4 bits altos del valor de EquipmentStatus del transponder, con los 4 bits altos de EquipmentStatus contenido en el comando SET de la fase de emisión del Recibo.
    - Después del paso anterior, pero solamente una vez por cada sesión de comunicaciones, el transponder incrementa el valor en los 12 bits bajos de EquipmentStatus.

6.9 MONTAJE DEL TRANSPONDER

Como una medida de seguridad, los transponders utilizados en las concesiones del MOPTT deberán instalarse en forma fija al vehículo. En el caso que ellos sean desmontados, deberán perder automáticamente su programación, o alternativamente

contar con algún otro sistema que permita detectar esta situación. Para informar un eventual desmontaje del transponder, se utilizará un bit en el Atributo obeStatus, de acuerdo a lo especificado en el Anexo B.

6.10 SOFTWARE DEL PUNTO DE COBRO

En la Transacción Nacional de Peaje Interoperable especificada en el Anexo A, siempre ocurren dos pasos de autenticación, el primero con una clave interoperable, y el segundo con una desconocida por la concesión que ejecuta la Transacción. Este esquema asegura que en la generación de la Transacción participó un transponder determinado, que se identifica a través del Atributo ContractSerialNumber. Además, los valores de autenticación acusan cualquier modificación posterior de datos fundamentales de la Transacción aportados por el transponder.

Sin embargo, una Transacción de peaje incluye además de los datos suministrados por el transponder, otros generados por el RSE. El software del punto de cobro es el encargado de ensamblar todos los datos en un registro de Transacción. El sistema de cobro deberá incorporar los medios para determinar posibles cambios introducidos a este software, que pudieran conducir a transacciones ensambladas en forma errónea.

6.11 RESGUARDO DE LAS CLAVES DE SEGURIDAD

En todos los puntos del sistema donde deban residir las claves, será obligatoria la aplicación de medidas de seguridad para su resguardo, con el fin de impedir que las claves pasen a poder de terceros. Estas medidas deberán ser propuestas por cada concesión, dentro de lo establecido en [MOPTT – ST3] y requerirán de la aprobación por parte del MOPTT.

Para el proceso de producción, el MOPTT facilitará a los fabricantes las claves de acceso MEAcK y todas las claves de autenticación de los grupos MEAuKF y MEAuKI. Sin embargo, y como una medida adicional de seguridad, el MOPTT distribuirá a las concesiones en primera instancia solamente las claves MEAcK y una del grupo MEAuKI. Las restantes claves maestras de autenticación serán liberadas por el MOPTT en la medida que ellas sean requeridas para mantener la seguridad del sistema.

La entrega de las claves de seguridad por parte el MOPTT se regirá por lo establecido en el capítulo 8, en [MOPTT – ST3] y en [MOPTT – ST6].

7 Interoperabilidad en las Concesiones del MOPTT

El presente documento establece la Transacción nacional estandarizada para el pago electrónico de peaje, que posibilita a un mismo transponder a bordo del velículo interactuar con todas las plazas de peaje de las concesiones del MOPTT. Esto significa que:

- El transponder único del vehículo utiliza comunicaciones DRSC estandarizadas para intercambiar datos con los equipos del punto de cobro.
- Se especifican los tipos de datos que intervienen en la Transacción y asimismo las funciones con las que se lleva a cabo su acceso.
- Todos los detalles de la Transacción están definidos, incluido el esquema de seguridad, por lo que las transacciones son procesadas correctamente en todos los puntos de cobro.

Esta plataforma ha sido concebida para permitir la interoperabilidad entre concesiones, de manera que los usuarios puedan suscribir un solo contrato para tener acceso al peaje electrónico, y recibir una cuenta consolidada por este servicio, sin importar si han utilizado las vías de una o de varias concesiones. El MOPTT recomienda a las diferentes concesiones establecer los necesarios acuerdos y convenios mutuos que sean requeridos para instaurar un sistema completamente integrado.

7.1 ESCENARIO DE CONCESIONES MÚLTIPLES

La figura 7.1 presenta el escenario que se genera en un ambiente de concesiones múltiples.

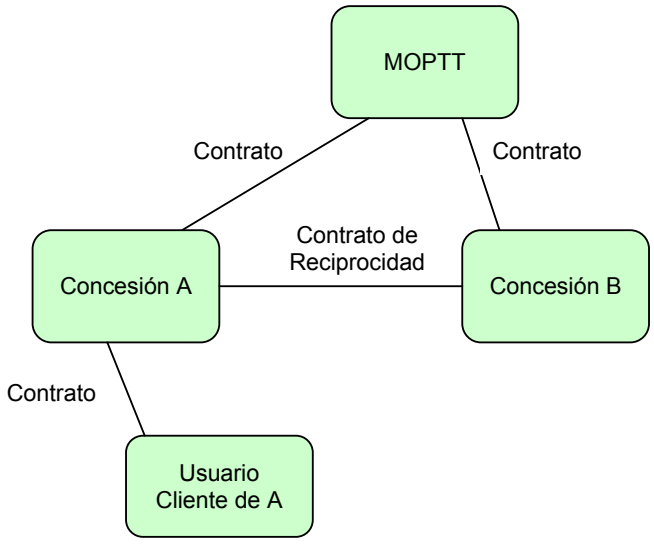

Figura 7.1 Escenario de Concesiones Múltiples

El transponder es emitido por la concesión “A”, después de que el usuario suscribe el correspondiente contrato de pago. La concesión “B” representa a cualquiera de las otras concesiones por las que circula el usuario. El MOPTT se relaciona con “A” y “B” a través de los respectivos contratos de concesión.

La existencia de una cuenta única al usuario requiere la suscripción de un contrato de reciprocidad entre “A” y “B”, el que es materia de acuerdo entre las partes. En este caso, “A” actúa como proveedor de servicio de pago para “B”, integrando los cobros de éste en la cuenta del usuario. Para este efecto, las Condiciones Generales de Uso de Televía que forman parte del convenio suscrito por el usuario, establecen que el cobro podrá ser realizado independientemente o conjuntamente por las sociedades concesionarias.

7.2 INFORMACIÓN COMPARTIDA

Para hacer posible la interoperabilidad, el MOPTT pone a disposición de todas las concesiones la siguiente información:

- La clave de acceso maestra MEAcK, generada por el MOPTT, que permite a los equipos del punto de cobro RSE acceder a los datos de la memoria del transponder, bajo las condiciones de acceso definidas en las Tablas 4.2 a 4.5.
- Las claves de autenticación maestras MEAuKI del dominio interoperable, que son usadas por el RSE para determinar si un transponder es auténtico.
- La lista de clientes de todas las concesiones, con los respectivos números de contrato.
- Otros antecedentes tales como características particulares de los modelos de transponder usados, sus Elementos de sistema, etc.

7.3 FLUJO DE DATOS DE SEGURIDAD

7.3.1 TRANSACCIÓN EJECUTADA POR “A”

La figura 7.2 muestra los flujos de los datos de seguridad que tienen lugar cuando la Transacción ocurre en “A”, es decir, en la concesión emisora del transponder. Los flujos responden a los siguientes requerimientos básicos:

- El acceso a los datos del transponder queda restringido a entidades autorizadas:

  “A” accede a la memoria del transponder gracias a que conoce la clave maestra MEAcK, con la que calcula la credencial de acceso dinámica AC_CR.

- “A” necesita una prueba de que el transponder que se comunica es válido, y que por lo tanto su tarifa le será pagada:

  Para este fin, el transponder genera el número de autenticación dinámico OBEAuthenticator, usando una clave del grupo EAuKA residente en el transponder. “A” verifica el valor de OBEAuthenticator usando la clave maestra MEAuKA de su propio dominio.

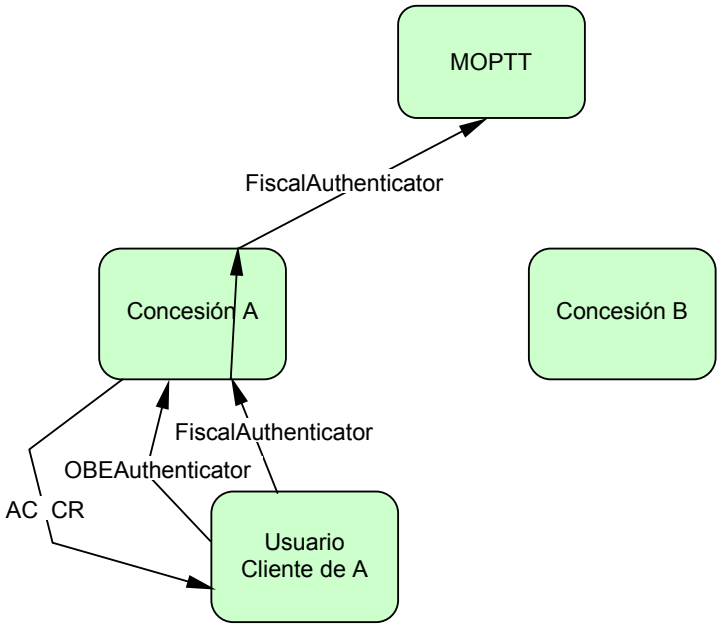

Figura 7.2 Flujo de Datos de Seguridad cuando la Transacción es ejecutada por "A"

- El MOPTT y también el usuario requieren una prueba de que la Transacción fue generada por "A" en forma legítima:

  El transponder genera el número de autenticación dinámico FiscalAuthenticator, usando una clave del grupo EAuKF residente en el transponder.

- El usuario no puede desconocer la Transacción:

  Gracias a la diversificación de las claves en los transponders, sólo es posible calcular el valor de FiscalAuthenticator con la participación de un transponder específico.

7.3.2 TRANSACCIÓN EJECUTADA POR "B", CON RECIPROCIDAD ENTRE "A" Y "B"

Algunos requerimientos cambian cuando la Transacción ocurre en "B", es decir, en una concesión diferente a la que emitió el transponder. En la figura 7.3, los flujos responden a los requerimientos:

- El acceso a los datos del transponder queda restringido a entidades autorizadas:

  "B" accede a la memoria del transponder gracias a que conoce la clave maestra MEAcK, con la que calcula la credencial de acceso dinámica AC_CR.

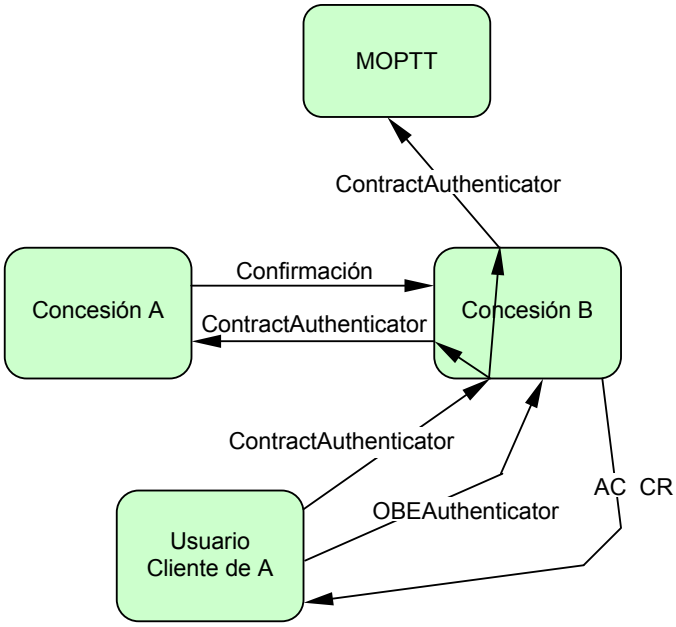

Figura 7.3 Flujo de Datos de Seguridad cuando la Transacción es ejecutada por “B”, con Reciprocidad entre “A” y “B”.

- “B” necesita una prueba de que el transponder que se comunica es válido, y que por lo tanto su tarifa le será pagada:

  Para este fin, el transponder genera el número de autenticación dinámico OBEAuthenticator, usando una clave del grupo EAuKI residente en el transponder.

- “A” y también el usuario necesitan una prueba que todo pago demandado por “B” es legítimo:

  El transponder genera un número de autenticación dinámico ContractAuthenticator, usando una clave de valor desconocido para “B”, del grupo EAuKA residente en el transponder, perteneciente al dominio de “A”.

- El usuario no puede desconocer la Transacción:

  Gracias a la diversificación de las claves en los transponders, sólo es posible calcular el valor de ContractAuthenticator con la participación de un transponder específico.

- “B” debe ser informado por “A” si el cargo ha sido aceptado o no:

  “A” envía a “B” una confirmación positiva o negativa dentro de un plazo preestablecido. Si “B” no recibe confirmación dentro del plazo convenido, “B” entiende que el cargo ha sido aceptado.

- Cuando “A” rechaza el cargo, se requiere un procedimiento para determinar si el rechazo es válido:

  El valor de autentificación dinámico ContractAuthenticator requiere para su verificación una entidad independiente en la que confíen las partes, la que debe contar con una unidad sellada SAM que contenga las claves maestras MEAuKA del dominio de “A”. Esta unidad recalcula internamente el valor de ContractAuthenticator, lo compara con el suministrado por “B” y emite una respuesta positiva o negativa. En la figura 7.3 esta entidad es el MOPTT.

7.3.3 TRANSACCIÓN EJECUTADA POR “B”, SIN RECIPROCIDAD ENTRE “A” Y “B”

Aunque no es la situación deseada, “A” y “B” pueden operar en forma independiente, sin acuerdo de reciprocidad entre ambos. En virtud del convenio suscrito por el cliente con “A” y las condiciones generales de uso del Televía, “B” no considera a dicho usuario como infractor. En cambio, genera y le envía el correspondiente documento de cobro. Estos usuarios reciben entonces varias cuentas por concepto de peaje. Los flujos de seguridad se muestran en la figura 7.4.

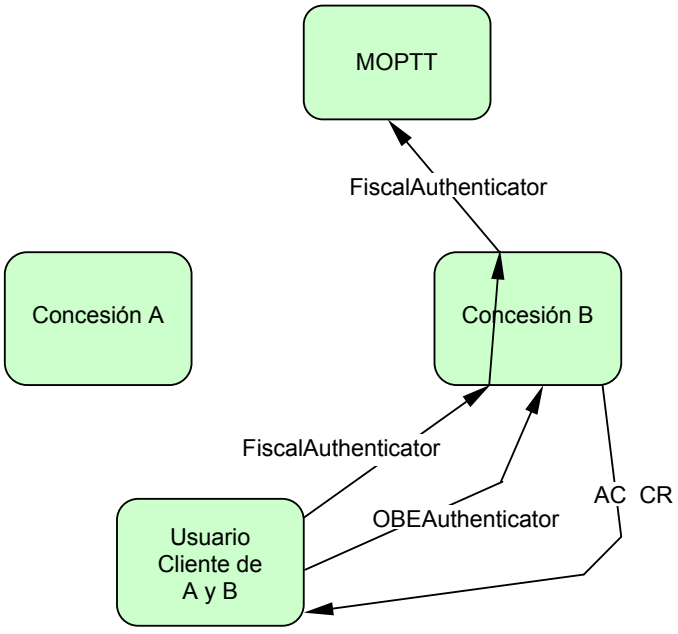

Figura 7.4 Flujo de Datos de Seguridad cuando la Transacción es ejecutada por “B”, sin Reciprocidad entre “A” y “B”

Cuando “B” cobra directamente a los usuarios, no tiene objeto generar el autenticador ContractAuthenticator como en 7.3.2, porque “B” no necesita probar nada ante “A”. En cambio, “B” calcula el valor dinámico FiscalAuthenticator, con lo que los flujos de

seguridad responden a requerimientos similares a los presentados en 7.3.1, pero esta vez referidos a la concesión “B”:

- El acceso a los datos del transponder queda restringido a entidades autorizadas:

  “B” accede a la memoria del transponder gracias a que conoce la clave maestra MEAcK, con la que calcula la credencial de acceso dinámica AC_CR.

- “B” necesita una prueba de que el transponder que se comunica es válido, y que por lo tanto su tarifa le será pagada:

  Para este fin, el transponder genera el número de autenticación dinámico OBEAuthenticator, usando una clave del grupo EAuKI residente en el transponder. “B” verifica el valor de OBEAuthenticator usando la clave maestra MEAuKI del dominio interoperable.

- El MOPTT y también el usuario requieren una prueba de que la Transacción fue generada por “B” en forma legítima:

  El transponder genera el número de autenticación dinámico FiscalAuthenticator, usando una clave del grupo EAuKF residente en el transponder.

- El usuario no puede desconocer la Transacción:

  Gracias a la diversificación de las claves en los transponders, sólo es posible calcular el valor de FiscalAuthenticator con la participación de un transponder específico.

7.4 LISTAS NEGRA, GRIS, AMARILLA Y VERDE

Está previsto que cuatro bits del Atributo EquipmentStatus del transponder sean utilizados para indicar que éste se encuentra en lista negra, gris, amarilla y/o verde respectivamente. Cada vez que un transponder pasa a integrar o sale de una de estas listas, la concesión que decide efectuar dicha acción es responsable de transmitir la información correspondiente al MOPTT, y de cambiar el estado del bit particular dentro del transponder. Opcionalmente, la concesión comunicará esta información a otras concesiones con las que tenga convenio de reciprocidad, de acuerdo a los términos establecidos en los respectivos convenios. En todo caso, las concesiones siempre pueden obtener esta información del transponder mismo o del MOPTT.

En general, solamente el emisor del transponder está autorizado para manipular los bits correspondientes a las listas negra, gris y verde. No obstante lo anterior, las concesiones podrán proponer al MOPTT esquemas de operación alternativos. Teniendo como objetivo mantener el ambiente de interoperabilidad, el MOPTT estudiará las respectivas propuestas, podrá convenir eventuales modificaciones, y comunicará su decisión a las partes interesadas, indicando en ella el plazo para su implantación. Si la solicitud es aprobada, se procederá según lo indicado en 2.3.

7.4.1 LISTA NEGRA

Cuando un transponder ha sido robado o se ha puesto término a un contrato sin devolución del transponder, el concesionario emisor “A” tiene la responsabilidad de colocar el transponder en lista negra. En la siguiente pasada por un punto de cobro del emisor del transponder, el RSE activa el bit de lista negra del transponder. Otras concesiones no están autorizadas para manipular este bit, a menos que ello esté previsto en un esquema alternativo de operación de las listas, aprobado por el MOPTT.

A partir del momento en que una concesión es informada de que un transponder está en lista negra, exige al vehículo afectado el pago en forma manual en todos los lugares de cobro donde existe barrera. Opcionalmente, puede dar cuenta a la policía de esta situación. En sistemas de flujo libre, el operador registra la imagen de la patente del vehículo, y aplica los procedimientos correspondientes al cobro de una infracción potencial.

Son materia del convenio de reciprocidad entre “A” y otra concesión “B”, los posibles pagos cruzados para los casos de lista negra. A modo de ejemplo, puede establecerse que “A” paga a “B” todas las transacciones válidas que ocurran en éste, hasta el instante en que “A” informa que un transponder pasa a condición de lista negra. Una medida de este tipo incentiva a mantener al día las listas negras.

El emisor del transponder tiene la responsabilidad de volver a cero el bit de lista negra una vez que el transponder ha sido recuperado. Otras concesiones sólo estarán autorizadas para efectuar esta acción, si ella está prevista en un esquema alternativo de operación de las listas, aprobado por el MOPTT.

7.4.2 LISTA GRIS

Cuando el emisor del transponder “A” rechaza un cobro presentado por “B”, existe una sospecha fundada de que algo anormal ocurre con el transponder involucrado. Por tratarse de uno de sus clientes, es responsabilidad de “A” informar al MOPTT esta situación, y de activar el bit de lista gris en el interior del transponder. Otras concesiones no están autorizadas para manipular este bit, a menos que ello esté previsto en un esquema alternativo de operación de las listas, aprobado por el MOPTT.

A partir del momento en que una concesión es informada de que un transponder está en lista gris, exige al vehículo afectado el pago en forma manual en todos los lugares de cobro donde existe barrera. En sistemas de flujo libre, el operador registra la imagen de la patente del vehículo, y aplica los procedimientos correspondientes al cobro de una infracción potencial.

Son materia del convenio de reciprocidad entre “A” y otra concesión “B”, los posibles pagos cruzados para los casos de lista gris. Si no existe acuerdo de reciprocidad con “A”, el operador decide la acción a tomar. Por ejemplo, si el usuario tiene un buen historial en sus pagos y la transacción se ha completado en forma satisfactoria, el operador puede descartar la imagen de la patente y cobrar en base a la transacción.

El emisor del transponder tiene la responsabilidad de volver a cero el bit de lista gris una vez que el problema ha sido aclarado. Otras concesiones sólo estarán autorizadas para efectuar esta acción, si ella está prevista en un esquema alternativo de operación de las listas, aprobado por el MOPTT.

7.4.3 LISTA AMARILLA

Cuando un usuario está excedido en todos los plazos para pagar su cuenta con una determinada concesión, la concesión afectada está autorizada para colocar el transponder en lista amarilla, activar el bit correspondiente del transponder e informar al MOPTT. Después de esto, la concesión involucrada trata al usuario como infractor. Desde el momento en que otras concesiones son informadas de que un transponder se encuentra en lista amarilla, tomarán acciones como las siguientes:

- En todos los lugares de cobro donde existe barrera, exigencia de pago manual al vehículo afectado.
- En sistemas de flujo libre, y como medida precautoria, registro de la imagen de la patente del vehículo, y aplicación de los procedimientos correspondientes al cobro de una infracción potencial. Acciones adicionales dependerán de la relación particular de cada concesión con el usuario.

El uso de la lista amarilla es optativo. Las concesiones que deseen usar esta lista, deberán acordar los procedimientos específicos que sean necesarios para manejar en forma correcta el bit amarillo en el interior del transponder, en un ambiente de operadores múltiples. Estos procedimientos deberán contar con la aprobación del MOPTT.

7.4.4 LISTA VERDE

Los transponders deberán ser entregados por la fábrica y distribuidos a los usuarios con el bit de lista verde activado. Solamente después de que se ha autorizado el uso de un determinado transponder, el emisor “A” coloca a dicho transponder en la lista verde. En la siguiente pasada por un punto de cobro del emisor, el RSE coloca en cero el bit de lista verde del transponder. Otras concesiones sólo estarán autorizadas para efectuar esta acción, si ella está prevista en un esquema alternativo de operación de las listas, aprobado por el MOPTT. Eventualmente, esta primera pasada puede ser usada para activar funciones adicionales, como por ejemplo la captura de una imagen para asegurar que el transponder está en el vehículo correcto.

De esta manera, el bit de lista verde en cero puede ser usado por cualquier concesión del MOPTT para reconocer si un determinado transponder ha sido autorizado para usar los sistemas de cobro electrónico, incluso en sistemas que actualizan las listas menos frecuentemente.

7.5 CLASIFICACIÓN DE VEHÍCULOS

El Atributo VehicleClass residente en el transponder, que corresponde a la clase declarada del vehículo, es el dato primario usado por el punto de cobro para identificar el tipo de vehículo.

En las concesiones del MOPTT, el esquema de clasificación para las concesiones urbanas es diferente al de las interurbanas, por lo que el valor a consignar en el Atributo VehicleClass depende del tipo de concesión que emite el transponder. Para facilitar la interoperabilidad, el MOPTT recomienda a las concesiones emisoras de transponder a grabar en VehicleClass ambos valores de clasificación, urbano e interurbano, conforme a las definiciones presentadas en el Anexo B.

Por otro lado, el sistema de clasificación del punto de cobro mide características físicas del vehículo, y determina si éstas son compatibles con la clase declarada, dentro de determinados márgenes de tolerancia. Cuando existe correspondencia, el RSE acepta como válida la clase declarada y con ella calcula la tarifa.

Si se produce una discrepancia, se procede de la manera siguiente:

- Si la clase declarada implica una tarifa más alta que la obtenida a partir de las mediciones efectuadas por el sistema de clasificación, se aplica la tarifa correspondiente a clase medida.

- Cuando la clase declarada conduce a una tarifa más baja que la obtenida a partir de las mediciones efectuadas por el sistema de clasificación, pueden producirse dos situaciones:

  - ✓ Un transponder ha sido trasladado a un vehículo más grande. Esto constituye una infracción potencial o una falla de equipo, y por lo tanto en sistemas de flujo libre es necesario registrar la imagen de la patente del vehículo. En sistemas provistos de barrera, se detiene el vehículo hasta que la situación se verifique en el lugar.
  - ✓ El vehículo ha variado su clase debido a que lleva un remolque. Puesto que ello no es una infracción, se aplica la tarifa correspondiente a la clase medida.

  El sistema electrónico de cobro, con el apoyo de los datos aportados por el sistema de clasificación, debe ser capaz de diferenciar ambas situaciones.

7.6 BITS R Y T EN OBESTATUS.OBECONFIGURATION

El bit R en obeStatus.OBEConfiguration tiene el valor 1 mientras el transponder se encuentra desmontado del vehículo, y pasa a 0 al ser montado. Esto permite al sistema RSE reconocer si un transponder está instalado en un vehículo. La existencia del bit R en un transponder es opcional.

En cambio, el bit T en obeStatus.OBEConfiguration queda en 1 luego de una manipulación ilegal del transponder, por ejemplo al ser desmontado del vehículo o al ser abierto. El emisor “A” está facultado para devolverlo a 0 en la siguiente pasada por uno de

sus puntos de cobro, si se cumplen las condiciones establecidas por la concesión y aprobadas por el MOPTT para llevar a cabo esta intervención. Otras concesiones sólo estarán autorizadas para efectuar esta acción, si ella está prevista en un esquema alternativo de operación de este bit, aprobado por el MOPTT. En la pasada mencionada se podrán activar funciones adicionales, como por ejemplo la captura de una imagen para verificar si el transponder se encuentra en el vehículo correcto.

7.7 FORMATO PARA REPRESENTAR EL CONTRACTSERIALNUMBER

El Atributo ContractSerialNumber es el único valor que permite identificar al cliente de una concesión en la transacción DSRC de peaje. Este Atributo ocupa 4 bytes y por lo tanto corresponde a un valor entre 0 y FF FF FF FF en notación hexadecimal. Para facilitar el ingreso de los transponders a los sistemas de las concesiones, el valor de ContractSerialNumber del Elemento de Cobro de Peaje Interoperable deberá estar impreso en la carcasa del transponder mediante 10 dígitos decimales más un dígito verificador, de acuerdo al formato que se entrega en B.2, junto a la codificación de dicho Atributo. Además, deberá estar impreso en la carcasa el mismo valor mediante código de barras, en formato CODE 128, para permitir una lectura mecanizada.

8 Registro de Antecedentes

El MOPTT registrará todos los antecedentes tales como parámetros, Atributos privados, valores específicos, listas de clientes, etc, que sean utilizados en las transacciones con los transponders de las concesiones del MOPTT.

Para este fin, los antecedentes respectivos deberán ser comunicados al MOPTT con la anticipación requerida, en medio digital.

Otros antecedentes como claves de seguridad, propias de cada concesión, deberán ser entregadas en custodia en un medio sellado, no alterable, con instrucciones de ser entregadas al MOPTT en caso de término de la concesión. La identidad de la notaría deberá ser comunicada al MOPTT.

Los antecedentes deberán incluir el nombre de la concesión, dirección de correo electrónico para formular consultas, y el nombre de la persona responsable de los datos.

Cuando los antecedentes requieran de una aprobación por parte del MOPTT, o cuando sea de conveniencia introducir modificaciones a lo que se propone en ellos, el MOPTT tomará contacto con el remitente para acordar los detalles.

El MOPTT colocará todos los antecedentes recopilados que puedan ser de conocimiento compartido a disposición de todas las concesiones.

Antecedentes como claves de seguridad de interoperabilidad que son definidas por el MOPTT, serán entregados en medios que sólo podrán ser interpretados por los sistemas. Esta información deberá ser solicitada por el representante legal de cada concesión, y será entregada previa firma del acuerdo de confidencialidad de la información suministrada.

9 Conformidad con la Presente Especificación

Todos los modelos de transponders que sean considerados para su distribución en las concesiones del MOPTT, deberán ser sometidos a un proceso de homologación de conformidad con lo establecido en el presente documento.

Para este fin, el documento [MOPTT – ST2] emitido por el Ministerio de Obras Públicas, Transportes y Telecomunicaciones de la República de Chile, especifica las diferentes pruebas a que debe ser sometido cada uno de los modelos de producción de los transponders considerados. Será responsabilidad de las respectivas concesiones que sus proveedores gestionen con organismos calificados la ejecución de dichas pruebas y obtengan un certificado de conformidad, el que deberá ser presentado al MOPTT.

Cada concesión deberá acordar con el MOPTT el procedimiento con el cual se determinará si una institución puede ser autorizada para ejecutar las pruebas, incluida la generación de la correspondiente “test suite”. Las instituciones propuestas deberán ser reconocidas como entidades certificadoras por el MOPTT y ningún caso podrán tener capitales relacionados con el o los proveedores.

Anexo A. Transacciones Reconocidas por el MOPTT

Este anexo presenta las transacciones reconocidas para los transponders de las concesiones del MOPTT. En la Tabla A.1 de la sección A.1 se definen las transacciones reconocidas, y a partir de la sección A.2 se encuentran las respectivas descripciones.

A.1 LISTA DE TRANSACCIONES RECONOCIDAS POR EL MOPTT

TABLA A.1 Transacciones Reconocidas por el MOPTT	
Transacción	***Identificación de la Aplicación (AID)***
Nacional de Peaje Interoperable	1 (EFC)
Gestión de Estacionamientos	6 (Parking-Management)
Sonda de Tráfico	29 (Private)

A.2 TRANSACCIÓN NACIONAL DE PEAJE INTEROPERABLE

La Transacción de referencia especificada a continuación ha sido concebida para satisfacer los requerimientos de interoperabilidad dentro del ambiente de las concesiones viales del Ministerio de Obras Públicas. Ella ha sido construida usando las herramientas provistas por [A1], adaptando soluciones planteadas en [CARDME – 3].

A.2.1 FASES DE LA TRANSACCIÓN

La Transacción se compone de las fases:

- Inicialización
- Núcleo de la Transacción:
  - ✓ Presentación
  - ✓ Cuando es un transponder nativo, o uno foráneo sin contrato de reciprocidad entre las concesiones involucradas:
    - Autenticación Fiscal
  - ✓ Cuando es un transponder foráneo con contrato de reciprocidad entre las concesiones involucradas:
    - Autenticación del Contrato
  - ✓ Recibo
- Término de la Transacción

A.2.2 DESARROLLO DE LA TRANSACCIÓN

El desarrollo de la Transacción de referencia se resume a continuación.

A.2.2.1 Inicialización

La figura A.2.1 presenta la fase de Inicialización. La parte de la respuesta VST marcada con el borde derecho sombreado es debida a la presencia del elemento reservado al emisor del transponder.

Fase	*RSE*		*OBE*
Inicialización	**INITIALISATION.request** (BST)	→	
		←	**PrivateWindowRequest** (PrWRq)
	PrivateWindowAllocation (PrWA)	→	
		←	**INITIALISATION.response** (VST) • AID = 1, EID = n1 [Peaje Interoperable] • EFC-ContextMark ▪ ContractProvider ▪ TypeOfContract ▪ ContextVersion • OBEGroupID • RndOBE • AID = 1, EID = n2 [Emisor del Transponder] • EFC-ContextMark ▪ ContractProvider ▪ TypeOfContract ▪ ContextVersion • OBEGroupID • RndOBE • ObeConfiguration: ▪ EquipmentClass ▪ ManufacturerID ▪ OBEStatus

Figura A.2.1 Inicialización de la Transacción.

A.2.2.2 Núcleo de la Transacción

A.2.2.2.1 Caso 1

La figura A.2.2 presenta el núcleo de la Transacción, cuando el usuario ha suscrito un contrato de pago con la concesión que ejecuta la Transacción. Este caso comprende las siguientes variantes:

- Caso 1A, transponder nativo: La Transacción es ejecutada en instalaciones de la concesión que emitió el transponder.
- Caso 1B, transponder foráneo sin reciprocidad: La Transacción se lleva a cabo en instalaciones de una concesión diferente a la que emitió el transponder, sin acuerdo de reciprocidad entre ambas.

Sólo en la primera variante son posibles las zonas optativas con las que el emisor del transponder accede a la información guardada en el Elemento reservado para él, o a la información del Elemento de Sistema.

A.2.2.2.2 Caso 2, Transponder Foráneo con Reciprocidad

La figura A.2.3 muestra el caso cuando la Transacción es ejecutada en instalaciones de una concesión diferente a la que emitió el transponder, y existe acuerdo de reciprocidad entre ambas.

Fase	*RSE*		*OBE*
Presentación ***(Optativo: Lectura de los Atributos Receipt Authenticator y Spare)***	**GET_STAMPED.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • EquipmentStatus [RndRSE, KeyRef_X] (1) **GET.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • ContractSerialNumber • ContractValidity • ReceiptServicePart • SessionClass • ReceiptAuthenticator • VehicleClass • Spare	→	
		←	**GET_STAMPED.response** • EquipmentStatus • OBEAuthenticator [Auth_X] (2) **GET. response** • ContractSerialNumber • ContractValidity • ReceiptServicePart • SessionClass • ReceiptAuthenticator • VehicleClass • Spare
Autenticación Fiscal ***(Optativo: Lectura del Elemento del Emisor y Lectura de Atributos del Elemento de Sistema)***	**GET_STAMPED.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • EquipmentStatus [RndRSE, KeyRef_F] **GET.request** [← EID = n2] • AC_CR [RndOBE, ElementAccessKey] • Scratchpad **GET.request** (3) [← EID = 0] • AC_CR [RndOBE, ElementAccessKey] • ActivityTimer • BatteryInsertionDate	→	
		←	**GET_STAMPED.response** • EquipmentStatus • FiscalAuthenticator [Auth_F] **GET.response** • Scratchpad **GET.response** (3) • ActivityTimer • BatteryInsertionDate
Recibo ***(Optativo: Escritura en los Atributos Receipt Authenticator y Spare, y Escritura en los Elementos del Emisor y del Sistema)***	**SET.request** [→ EID = n1] • AC_CR [RndOBE, ElementAccessKey] • ReceiptServicePart • SessionClass • ReceiptAuthenticator [DeReAuKey] • EquipmentStatus • Spare **SET.Request** [→ EID = n2] • AC_CR [RndOBE, ElementAccessKey] • Scratchpad **SET.Request** [→ EID = 0] • AC_CR [RndOBE, ElementAccessKey] • obeStatus **SET_MMI.request**	→	
		←	**SET.response** **SET.response** **SET.response** **SET_MMI.response**

Ver Notas 1 a 3 en la página siguiente

Figura A.2.2 Núcleo de la Transacción Nacional de Peaje Interoperable
(Casos 1A: Transponder nativo y 1B: Transponder foráneo sin reciprocidad)

Notas a la figura A.2.2:

1 En el caso 1A se usa para KeyRef_X el valor de KeyRef_A: 111_{10} o 112_{10}, y en el caso 1B se usa KeyRef_I: 115_{10} a 118_{10}.
2 En el caso 1A el transponder genera OBEAutenticator(Auth_A), y en el caso 1B el transponder genera OBEAutenticator(Auth_I)
3 En la configuración 4 de AIAs, el segundo comando GET.request de la fase de Autenticación Fiscal es diferente. Para leer el valor del atributo ActivityTimer se usa un comando privado, y para obtener el valor de BatteryInsertionDate se lee con un GET.request el atributo privado 125, que se encuentra en el Elemento con EID = 2. Los detalles se entregan en C.2.5. La respuesta cambia correspondientemente, según se presenta en C.2.6 y C.2.7.

Fase	*RSE*		*OBE*
Presentación ***(Optativo: Lectura de los Atributos Receipt Authenticator y Spare)***	**GET_STAMPED.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • EquipmentStatus [RndRSE, KeyRef_I] **GET.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • ContractSerialNumber • ContractValidity • ReceiptServicePart • SessionClass • ReceiptAuthenticator • VehicleClass • Spare	→	
		←	**GET_STAMPED.response** • EquipmentStatus • OBEAuthenticator [Auth_I] **GET. response** • ContractSerialNumber • ContractValidity • ReceiptServicePart • SessionClass • ReceiptAuthenticator • VehicleClass • Spare
Autenticación del Contrato	**GET_STAMPED.request** [← EID = n1] • AC_CR [RndOBE, ElementAccessKey] • EquipmentStatus [RndRSE, KeyRef_A]	→	
		←	**GET_STAMPED.response** • EquipmentStatus • ContractAuthenticator [Auth_A]
Recibo ***(Optativo: Escritura en Atributos Receipt Authenticator y Spare)***	**SET.request** [→ EID = n1] • AC_CR [RndOBE, ElementAccessKey] • ReceiptServicePart • SessionClass • ReceiptAuthenticator [DeReAuKey] • EquipmentStatus • Spare **SET_MMI.request**	→	
		←	**SET.response** **Set_MMI.response**

Figura A.2.3 Núcleo de la Transacción Nacional de Peaje Interoperable
(Caso 2: Transponder foráneo con reciprocidad entre concesiones involucradas)

A.2.2.3 Término de la Transacción

Fase	*RSE*		*OBE*
Tracking	**ECHO.request**	→	
		←	**ECHO.response**
	• • •	→ ←	
Cierre	**EVENT_REPORT.request** (Release)	→	

Figura A.2.4 Término de la Transacción.

A.2.3 DETALLES DE LA TRANSACCIÓN

A.2.3.1 Inicialización.

Cuando un transponder OBE ingresa a la zona de comunicaciones de un punto de cobro o RSE, capta una señal denominada BST, emitida por éste en forma periódica. En el presente caso, BST indica que la aplicación residente en el RSE es el cobro de peaje o EFC, cuyo código de identificación AID es igual a 1.

Luego de solicitar y recibir una ventana de comunicación como se explica en 5.1, el transponder emite la VST, que en esta oportunidad incluye una lista con dos aplicaciones EFC, correspondientes a los Elementos interoperable y el reservado al emisor del transponder. Cada componente de la lista incluye:

- El identificador de la aplicación AID, igual a 1 para EFC.
- El número identificador EID del Elemento en que residen los Atributos.
- El Atributo EFC-ContextMark, que identifica la entidad emisora del contrato residente en el OBE, el tipo de contrato y la versión de éste.
- El Atributo OBEGroupID, usado para diversificar la clave de acceso, como se describe en [A1].
- El valor aleatorio RndOBE, usado por el RSE para generar más adelante la credencial que autoriza el acceso al Elemento identificado previamente.

Después de la lista de aplicaciones, completan la VST los Atributos EquipmentClass, ManufacturerID y OBEStatus.

Gracias a que el número de serie del contrato con el usuario no está presente en la VST, el usuario permanece anónimo durante la fase de inicialización. Esta característica impide a entidades que ignoran la clave de acceso conocer quién es el usuario.

A.2.3.2 Núcleo de la Transacción

A.2.3.2.1 General

El núcleo de la Transacción contiene, además de los pasos de lectura y escritura de datos en el transponder, dos autenticaciones separadas. La primera, incluida en la fase de presentación, sirve para determinar si se trata de un transponder legítimo. La segunda, que según el caso puede ser fiscal o del contrato, se lleva a cabo con una clave residente en el transponder pero desconocida para la concesión en la que se genera la Transacción. De esta manera, las transacciones sólo pueden ser generadas con la presencia de un transponder válido. Es importante indicar que aunque todas las autenticaciones se calculan usando el Elemento EquipmentStatus, los Atributos ContractProvider y ContractSerialNumber quedan implícitamente protegidos, puesto que ellos intervienen en la diversificación de las claves de autenticación que se encuentran grabadas en el transponder.

A.2.3.2.2 Presentación

En la fase de presentación, el RSE solicita al transponder mediante la función Get_Stamped, el Atributo EquipmentStatus del Elemento interoperable. Puesto que el acceso a este Atributo se encuentra protegido, Get_Stamped incluye una credencial de acceso dinámica AC_CR calculada por RSE, usando el valor aleatorio RndOBE enviado previamente en la VST. Ya que todos los operadores del sistema de concesiones del MOPTT deben tener acceso a los Atributos del cobro de peaje interoperable, la clave para determinar la credencial de acceso es compartida.

La respuesta del transponder contiene el valor de EquipmentStatus, seguido de un autenticador OBEAuthenticator calculado en forma dinámica por el transponder. En este cálculo interviene el valor aleatorio RndRSE y la referencia KeyRef, ambos incluidos por RSE en el comando Get_Stamped. Mediante OBEAuthenticator el operador determina si se trata de un transponder válido, ya que transponders falsos desconocen el valor de la clave interna para efectuar el cálculo.

La clave de autenticación que emplea el OBE se especifica mediante el valor presentado en KeyRef. Para transponders nativos, el operador preferirá usar las claves de su propio dominio, derivadas de sus claves maestras MEAuKA. En cambio, cuando se trata de transponders foráneos, necesariamente deberá usar una clave interoperable, derivada de las maestras MEAuKI, conocidas por todas las concesiones. RSE identifica al emisor del transponder mediante el contenido del Atributo EFC-ContextMark.

En la fase de presentación, RSE usa además un comando GET para solicitar al transponder los valores de los Atributos del Elemento interoperable:

- ContractSerialNumber: identifica la cuenta del usuario en el Sistema de Atención de Clientes.
- ContractValidity: indica las condiciones de validez del contrato. Cuando se trate de un contrato de duración indefinida, caso típico para cuenta central de clientes, en el valor de ContractExpiryDate que forma parte de ContractValidity se anotará el valor 00 00.

- ReceiptServicePart: breve resumen de la Transacción previa, usada en sistemas de peaje cerrado como ticket de entrada.
- Session Class: registra la clase de vehículo aplicada durante la Transacción previa, además de la clase declarada en el transponder.
- ReceiptAuthenticator: autenticador, calculado con una clave propia de cada concesión, residente en los RSE, que permite al operador detectar si ReceiptServicePart o SessionClass de la transacción previa han sido adulterados. El uso de este Atributo en la transacción es optativo, ya que solamente tiene justificación en sistemas de peaje de tipo cerrado. En todo caso, el transponder deberá configurarse con ReceiptAuthenticator incluido, para permitir su eventual uso por otras concesiones. Además, cada concesión deberá informar al MOPTT si utiliza o no este Atributo en sus transacciones.
- VehicleClass: clase declarada del vehículo en que está montado el transponder. Cuando el vehículo lleve un remolque, el sistema de clasificación determinará la clase efectiva.
- Spare: el uso de este Atributo es optativo y queda sujeto a la aprobación por parte del MOPTT. A modo de ejemplo, puede ser usado para guardar un segundo recibo, correspondiente a una transacción anterior.

La información anterior interviene en la generación del registro de la transacción que construye el RSE, y una parte de ella se emplea en el cálculo de la tarifa a aplicar.

A.2.3.2.3 Autenticación Fiscal

Esta fase se aplica únicamente a los Casos 1A y 1B definidos en A.2.2.2.1. El transponder genera un valor de autenticación FiscalAuthenticator, usando una clave interna derivada de una de las claves maestras del MOPTT MEAuKF. Puesto que esta última no es conocida por ninguna de las concesiones, ellas no pueden por su cuenta calcular ni tampoco comprobar dicho valor. FiscalAuthenticator demuestra al MOPTT que la Transacción (al menos algunos datos importantes de ella) no pudo ser generada en forma autónoma por quien ejecuta la transacción. Este esquema protege también al usuario, al existir la garantía de que su transponder tuvo que estar presente para que se pudiera generar un FiscalAuthenticator válido. Al mismo tiempo, sirve de prueba de que el transponder efectivamente participó en la Transacción.

La fase de Autenticación Fiscal, y sólo cuando se trata de un transponder nativo, la Transacción puede incluir en forma opcional la lectura de la información guardada en el Elemento reservado al emisor del transponder. La utilización de esta opción queda sujeta a la aprobación por parte del MOPTT.

De la misma manera, en la fase de Autenticación Fiscal, y nuevamente sólo si se trata de un transponder nativo, la Transacción puede incluir comandos para leer la información de Atributos contenidos en el Elemento de sistema. Ejemplos de estos Atributos, propios de cada fabricante, son la fecha de instalación de la batería, el tiempo en que el transponder ha estado activo, etc. Ellos son útiles para la gestión del transponder por el operador que lo ha emitido.

A.2.3.2.4 Autenticación del Contrato

Esta fase se aplica únicamente al Caso 2 definido en A.2.2.2.2, con transponders foráneos y si existe acuerdo de reciprocidad entre ambas concesiones. El transponder genera un valor de autenticación ContractAuthenticator, usando una clave interna derivada de una de las claves maestras del emisor del transponder MEAuKA. Puesto que esta última no es conocida por las otras concesiones, ellas no pueden calcular ni tampoco comprobar dicho valor. ContractAuthenticator prueba al emisor del transponder que la Transacción (al menos algunos datos importantes de ella) sólo pudo haber sido generada con la participación del transponder en cuestión. Para cobrar la tarifa, la concesión que ejecuta la transacción transmite al emisor del transponder el informe de la Transacción, acompañado del ContractAuthenticator. Cuando el emisor acepta un cobro, queda implícito que se trataba de una Transacción auténtica, la que posteriormente no puede ser desconocida por el usuario.

A.2.3.2.5 Recibo

En esta fase, el RSE copia al transponder los detalles de la Transacción, tales como identificación del punto de cobro, pista, fecha y hora, etc. El comando de escritura SET incluye una credencial de acceso AC_CR, idéntica a la usada en la fase de presentación. Los Atributos transferidos al transponder son:

- ReceiptServicePart: sirve como ticket de entrada cuando es leído en la fase de presentación de la siguiente Transacción.
- SessionClass: contiene el valor de la clase efectiva aplicada durante la transacción.
- ReceiptAuthenticator: autenticador, que permite detectar si ReceiptServicePart o SessionClass han sido adulterados. Es calculado por RSE usando la clave DeReAuKey, propia de cada concesión. Su uso es optativo, de acuerdo a lo señalado en A.2.3.2.2.
- EquipmentStatus: contiene datos para la gestión de la aplicación EFC, según se especifica en las secciones 6.8 y 7.4.

En la fase de recibo, y sólo cuando la Transacción es ejecutada por el emisor del transponder, ésta puede incluir comandos para escribir información en el Elemento de sistema y/o en el reservado al emisor. El uso de esta opción queda sujeto a la aprobación por parte del MOPTT.

Concatenado con el o los comandos anteriores, RSE envía al transponder un comando SET_MMI, que avisa al conductor el resultado de la Transacción. De acuerdo con [ISO - EFC], el significado del parámetro de SET_MMI es:

- 0: Transacción normal
- 1: Transacción anormal
- 2: Contactar al operador

A.2.3.3 Término de la Transacción

La fase de “tracking” es optativa, y se emplea solamente en sistemas de peaje de flujo libre, cuando el RSE requiere efectuar un seguimiento del desplazamiento del transponder. Para este fin se usa el comando ECHO descrito en [ISO - EFC], con el

parámetro de largo igual a cero. Mediante este artificio se extiende el período de comunicaciones con el fin de capturar nuevos puntos de localización del transponder. La Transacción termina cuando el transponder abandona la zona de comunicaciones, o alternativamente mediante el comando EVENT_REPORT. request[Release].

Cuando no se efectúa la fase de tracking, la Transacción se cierra enviando al transponder el comando EVENT_REPORT. request[Release].

A.3 TRANSACCIÓN PARA GESTIÓN DE ESTACIONAMIENTOS

La Transacción se compone de las fases:

- Inicialización
- Presentación
- Recibo
- Cierre

El desarrollo de la Transacción se resume a continuación.

Fase	*RSE*		*OBE*
Inicialización	**INITIALISATION.request** (BST)	→	
		←	**PrivateWindowRequest** (PrWRq)
	PrivateWindowAllocation (PrWA)	→	
		←	**INITIALISATION.response** (VST) • AID = 6_{10}, EID • PM-ContextMark ▪ ContractProvider ▪ TypeOfContract ▪ ContextVersion • OBEGroupID • RndOBE • ObeConfiguration: ▪ EquipmentClass ▪ ManufacturerID ▪ OBEStatus
Presentación	**GET_STAMPED.request** [← EID = n3] • AC_CR [RndOBE, ElementAccessKey] • ContractSerialNumber [RndRSE, KeyRef]	→	
		←	**GET_STAMPED.response** • ContractSerialNumber • ContractAuthenticator
Recibo	**SET_MMI.request**	→	
		←	**SET_MMI.response**
Cierre	**EVENT_REPORT.request** (Release)	→	

Figura A.3.1 Transacción para Gestión de Estacionamientos.

A.3.1 OPERACIÓN

En la fase de Presentación, el RSE identifica al usuario a través del PM-ContextMark y del número de contrato. La transacción ofrece seguridad de acceso, de manera que solamente puede ser ejecutada por operadores con conocimiento de la correspondiente clave de acceso. Asimismo, el valor de ContractAuthenticator que garantiza la legitimidad del transponder, sólo puede ser verificado con el concurso de la clave maestra de autenticación aplicable. En la fase de recibo, el usuario es informado si la transacción ha sido exitosa.

A.4 TRANSACCIÓN PARA SONDA DE TRÁFICO

La Transacción se compone de las fases:

- Inicialización
- Presentación
- Actualización
- Cierre

El desarrollo de la Transacción se resume a continuación.

Fase	*RSE*		*OBE*
Inicialización	**INITIALISATION.request** (BST)	→	
		←	**PrivateWindowRequest** (PrWRq)
	PrivateWindowAllocation (PrWA)	→	
		←	**INITIALISATION.response** (VST) • AID = 29_{10}, EID • Private-ContextMark ▪ ContractProvider ▪ TypeOfContract ▪ ContextVersion • OBEGroupID • RndOBE • ObeConfiguration: ▪ EquipmentClass ▪ ManufacturerID ▪ OBEStatus
Presentación	**GET.request**[← EID = n4] • AC_CR [RndOBE, ElementAccessKey] • TemporayID	→	
		←	**GET. Response** • TemporaryID
Actualización	**SET.request**[← EID = n4] • AC_CR [RndOBE, ElementAccessKey] • TemporaryID	→	
		←	**SET. Response**
Cierre	**EVENT_REPORT.request** (Release)	→	

Figura A.4.1 Transacción para Sonda de Tráfico.

A.4.1 OPERACIÓN

Esta transacción permite determinar los tiempos de tránsito ente dos puntos en los que se instalan equipos interrogadores. La transacción ofrece seguridad de acceso, de manera que solamente puede ser ejecutada por operadores con conocimiento de la correspondiente clave de acceso. Para mantener el anonimato del usuario, mientras no se efectúen mediciones, los transponders llevan el valor cero en el Atributo TemporaryID.

Cuando un vehículo provisto de transponder pasa por un punto de medición, el RSE determina en la fase de Presentación el valor de TemporaryID. Existen dos posibilidades:

- TemporaryID igual a cero:
  En este caso, el RSE genera un número aleatorio, que transfiere en la fase de Actualización al Atributo TemporaryID del transponder, creando de esta manera una identificación transitoria en éste. En paralelo, el RSE agrega a su base de datos un registro de transacción con el valor de TemporaryID, la fecha y hora, y la localización del punto de medición.

- TemporaryID distinto de cero:
  El valor de TemporaryID corresponde a la identificación transitoria grabada en la transacción previa. El RSE registra el valor de TemporaryID, la fecha y hora, y la localización del punto de medición. Luego, en la fase de Actualización, el RSE transfiere el valor cero al Atributo TemporaryID, borrando la identificación transitoria del transponder. En el caso de que el punto de medición no sea el último de una cadena, se podrá grabar en lugar del valor cero un nuevo número aleatorio, el que se agrega al registro de transacción.

El sistema central correlaciona los registros captados en los diferentes puntos de medición, y determina los tiempos de tránsito entre ellos.

Anexo B. Codificación de los Atributos

B.1 ATRIBUTOS INDEPENDIENTES DE LA APLICACIÓN

Los Atributos Independientes de la Aplicación o AIAs describen las características esenciales del transponder. Aunque es natural que integren el Elemento de sistema, cada fabricante decide si los incluye en él o no. El MOPTT ha estandarizado cuatro configuraciones de AIAs, que se presentan en el Anexo F.

Atributo: OBEConfiguration

Configuración del transponder, según se define en [CEN – L7]. Acceso: no está permitido el acceso directo, este Atributo se transmite en la VST. No obstante lo anterior, en las configuraciones 1 a 3 de AIAs, los datos elementales de este Atributo pueden ser leídos en forma separada, mediante accesos al elemento de sistema. En las configuraciones mencionadas, también es posible modificar el estado del bit T de OBEStatus, que para este fin posee acceso de Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**EquipmentClass**	0 a 32767, definido por el fabricante del transponder	xxx xxxx	Significado debe registrarse en el MOPTT
2	Especifica capacidades y limitaciones del transponder		xxxx xxxx	
3	**ManufacturerId**	1: Kapsch Telecom	mmmm mmmm	Ver otras asignaciones en sitio de NEN, en http://www.nen.nl/cen278
4	Número de identificación del fabricante	2: Alcatel CGA 3: Combitech (*) 4: CSSI (*) 6: Q-Free (*) 7: Siemens 8: Thomson-CSF 9: Denso 10: Mitsubishi (Heavy Ind.) 11: MELCO 12: Toshiba 13: OKI … 28: Telvent (*) etc.	mmmm mmmm	Se indican con (*) los fabricantes de transponders certificados conforme a MOPTT-ST2.
5	**obeStatus** 16 “flags” para indicar estado del transponder	Ausencia de tarjeta ICC	A	Especificado en [GSS – 2.0]. En las configuraciones 1 a 4 estandarizadas por el MOPTT, no son usados los bits A, I, B, P. Se requiere autorización previa del MOPTT para usar uno o más de ellos. Bit T en 1: Transponder ha sido manipulado o removido del veh
		ICC no reconocida	I	
		Falla de Batería	B	
		Error en Interfaz c/Periférico	P	
		Manipulación ilegal del OBE	T	
		Último estado antes de pasar a estado de reposo: 000: BLOCKED 001: WAIT 010: INIT 011: READY 100: DATA	xxx	
6		Reservado para uso privado	ssss sss	En caso de usarse, el significado de estos bits debe registrarse en el MOPTT
			R	1 = Transponder está removido del vehículo. Este bit no es usado en la configuración 4.

Atributo: OBEGroupID

Identifica el grupo al que pertenece el transponder.
En las configuraciones 1 a 3 de AIAs, AttrID: 17_{10}. Acceso: sólo lectura.
En la configuración 4 de AIAs, AttrID: 37_{10}. Acceso: sólo lectura sin credencial de acceso.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		Número aleatorio uniformemente distribuido en el rango de 0 a 2047	0000 0ggg	Para 1 millón de transponders, cada grupo tiene cerca de 500 transponders
2			gggg gggg	

Atributos: ManufacturingSerialNumber / ManufacturerSerialNumber

AttrID: 1_{10}. Ambos Atributos se diferencian sólo en el nombre. El primero de ellos se emplea en la configuración 1 y el segundo en las configuraciones 2 y 3. Corresponde al número de serie de fabricación del transponder. Acceso: sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a $4.294.967.295_{10}$	aaaa aaaa	Año de producción, dos dígitos en código BCD
2			ssss ss	Semana de producción, binario 1 a 52
			nn	Número de serie único dentro de la semana, binario 1 a 262.143
3			nnnn nnnn	
4			nnnn nnnn	

Atributo: 125 (Privado)

AttrID: 125_{10}. Este es un Atributo privado que sólo es utilizado en la configuración 4. Combina los datos elementales de TransponderSerialNumber y BatteryInsertionDate. Acceso: sólo lectura sin credencial de acceso.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**TransponderSerialNumber**	0 a $4.294.967.295_{10}$	nnnn nnnn	Corresponde a un número decimal de 10 dígitos. Los 2 dígitos izquierdos son el tipo de transponder, y los 8 dígitos derechos el Nº de serie de ese tipo. Ejemplo: tipo = 10 y Nº = 234600 → TSN = 1002346100_{10} = 3B BE 96 74_h
2			nnnn nnnn	
3			nnnn nnnn	
4			nnnn nnnn	
5	**BatteryInsertionDate**	0 a 65.535_{10}	aaaa aaa	Año de inserción, 0 a 127
			s	Semana de inserción, 1 a 52
6			ssss s	Ejemplo: 08 B8h → semana 23 del año 2004
			rrr	No usados

BatteryInsertionDate de la configuración 4 de AIAs deberá ser convertido en el punto de cobro al formato de BatteryInsertionDateMOP, descrito en la pág. 59.

Atributo: ActivityTimer

Este Atributo lleva la cuenta del tiempo que el transponder ha estado activo. Acceso: sólo lectura.

En las configuraciones 1, 2 y 3, ActivityTimer ocupa 4 bytes y tiene AttrID igual a 7_{10}:

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a $4.294.967.295_{10}$	tttt tttt	La cuenta se lleva en unidades de tiempo, diferentes para cada configuración: Configuración 1: 1 ms, Configuración 2: 100 ms Configuración 3: 0.512 ms
2			tttt tttt	
3			tttt tttt	
4			tttt tttt	

En la configuración 4, ActivityTimer ocupa 6 bytes:

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a $281.474.976.710.655_{10}$	tttt tttt	La unidad de tiempo usada en la cuenta es de 0.5 µs
2			tttt tttt	
3			tttt tttt	
4			tttt tttt	
5			tttt tttt	
6			tttt tttt	

Por la disparidad de definiciones para ActivityTimer, el MOPTT ha definido el atributo ActivityTimerMOP, de 3 bytes, con la cuenta de los segundos que el transponder ha estado activo. Los ActivityTimer de las diferentes configuraciones de AIAs deberán convertirse en el punto de cobro a ActivityTimerMOP, redondeando sus valores al segundo más cercano. El valor de ActivityTimerMOP es usado por los niveles superiores del sistema de cobro en la gestión de los transponders.

Atributos: BatteryInsertionDate / BatteryInsertionDateMOP

AttrID: 16_{10}. Este Atributo permite determinar la fecha en que se instaló la batería en el transponder. Tiene la misma definición en las configuraciones 1, 2 y 3 [(1)]. Acceso: sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 65.535_{10}	dddd dddd	El valor representa el número de días desde el 1 de enero de 1970
2			dddd dddd	

[1] La configuración 4 maneja esta información en el Atributo privado 125

Atributo: NumberOfWake-ups

AttrID: 18_{10}. Este Atributo lleva la cuenta de las veces que el transponder se ha activado. Sólo es usado en la configuración 3. Acceso: sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 65.535_{10}	wwww wwww	El valor corresponde a la cantidad de veces que el transponder se ha activado
2			wwww wwww	

Atributo: NumberOfReleases

AttrID: 19_{10}. Este Atributo lleva la cuenta de las veces que la transacción ha terminado con un Release. Sólo es usado en la configuración 3. Acceso: sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 65.535_{10}	rrrr rrrr	El valor corresponde a la cantidad de veces que la transacción ha terminado con un Release
2			rrrr rrrr	

Atributo: NumberOfVSTs

AttrID: 20_{10}. Este Atributo lleva la cuenta de las veces que el transponder ha emitido una VST. Sólo es usado en la configuración 3. Acceso: sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 65.535_{10}	rrrr rrrr	El valor corresponde a la cantidad de veces que el transponder ha emitido una VST
2			rrrr rrrr	

Otros Atributos Privados definidos por el Fabricante

Podrán existir otros Atributos independientes de la aplicación de tipo privado, relativos a la programación y personalización del transponder, pero ellos no se usarán en las transacciones de las diferentes aplicaciones.

B.2 ATRIBUTOS DEL ELEMENTO DE COBRO DE PEAJE INTEROPERABLE

Atributo: EFC-ContextMark
AttrID = 0_{10}. Denota el contexto específico de cobro de peaje. Incluye la concesión emisora del transponder, el tipo de contrato y la versión de éste. El formato de este Atributo se encuentra definido en [ISO - EFC]. Acceso: no está permitido el acceso directo, este Atributo se transmite en la VST.

El MOPTT llevará en el documento complementario [MOPTT-ST1-1] un registro actualizado de los valores vigentes de este EFC-ContextMark, y cuidará que en el dominio de cada concesión no existan valores de EFC-ContextMark duplicados, ni coincidentes con los ApplicationContextMark de otras aplicaciones. El documento [MOPTT-ST1-1], en su última versión, se distribuirá a las concesiones cada vez que sea modificado.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**ContractProvider**			
1	***CountryCode***	457_{10}	0111 0010	CountryCode según ISO 3166-1:
2	Identificador del país		01	Chile = “CL”
2	***IssuerIdentifier***	Los números de identificación de las concesiones del MOPTT se indican en el Anexo D.	CC CCCC	El número de la concesión es asignado por el MOPTT.
3	Número de identificación de la concesión emisora del transponder.		CCCC CCCC	
4	**TypeOfContract**	Definido por el MOPTT	TTTT TTTT	El MOPTT definirá el valor a usar, el que cambiará con cualquiera de los siguientes sucesos: ▪ Cada nueva versión de transacción aprobada por el MOPTT ▪ Cada forma de contrato con los usuarios aprobada por el MOPTT ▪ Diferentes tipos de OBEs, de variados fabricantes, dentro de una misma concesión
5	Designación de las reglas que el “ContractProvider” aplica al contrato.		TTTT TTTT	
6	**ContextVersion** Versión de implementación	Definido por el MOPTT	VVVV VVVV	Identifica versión de claves de seguridad. Su valor es definido por el MOPTT.

Atributo: ContractSerialNumber
AttrID = 1_{10}. Número de serie que designa el contrato individual con el usuario. Su valor se asignará de acuerdo a lo indicado en la columna de comentarios. El formato de este Atributo está definido en [ISO - EFC]. Acceso: Sólo lectura. La lista de clientes, incluido el respectivo "ContractSerialNumber", debe registrarse en el MOPTT.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a $4.294.967.295_{10}$	aaaa aaaa	2 dígitos BCD: año de producción o programación, indexado por proveedor, 00 a 99. 4 bits altos: decenas, 4 bits bajos: unidades
2			ssss ss	Semana de producción o programación dentro del año, binario 1 a máximo de semanas en el año
			nn	Número de serie único dentro de la semana de producción o programación
3			nnnn nnnn	
4			nnnn nnnn	

Cuando una concesión distribuya transponders de más de un proveedor, deberá indexar el año de producción o programación del transponder según la ecuación siguiente:

Año = [(año de producción o programación) + 10*n] módulo 100

en que n: entero entre 0 y 9, diferente para cada proveedor. Cada concesión deberá aplicar los resguardos necesarios para que no existan "ContractSerialNumber" duplicados.

Para la impresión del valor de ContractSerialNumber en la carcasa del transponder, se usará el formato que se especifica a continuación. Para este fin se divide ContractSerialNumber en cuatro segmentos:

Segmento 1: 4 bits altos Segmento 2: 4 bits siguientes
Segmento 3: 6 bits siguientes Segmento 4: 18 bits finales

El formato de impresión es el siguiente:

Dígito 1 (izquierdo): Valor del segmento 1 Dígito 2: Valor del segmento 2
Dígitos 3 al 4: Valor del segmento 3 Dígitos 5 al 10: Valor del segmento 4
Dígito 11: Código de verificación Luhn

Ejemplo: ContractSerialNumber: $12\ 34\ 56\ 78_{16}$

Segmento 1: $0001_2 = 1_{10}$
Segmento 2: $0010_2 = 2_{10}$
Segmento 3: $0011\ 01_2 = 13_{10}$
Segmento 4: $00\ 0101\ 0110\ 0111\ 1000_2 = 022136_{10}$

En formato de impresión, resulta: 1213022136L

Atributo: ContractValidity

AttrID = 2_{10}. Denota condiciones específicas para la validez del contrato. Acceso: Sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**ContractRestrictions**		RRRR RRRR	Permite definir restricciones más específicas a la validez, adicionales a las contenidas en TypeOfContract que se incluye en las VST. Definiciones usadas deben registrarse en el MOPTT.
2	Codificación específica del "ContractProvider", de las restricciones a la validez de una cuenta.		RRRR RRRR	
3			RRRR RRRR	
4			RRRR RRRR	
5	**ContractExpiryDate**	01.01.1990 a 31.12.2117	yyyy yyym	Tipo DateCompact, especificado en [ISO - EFC]: yyyyyyy: año 1990 (0) a 2117 (127) mmmm: mes (0..12); dddd: día (0..31). Todos los bits en 0 se usan para representar "sin fecha de término".
6	Contrato vence a las 24h de la fecha ContractExpiryDate		mmmd dddd	

Atributo: ReceiptServicePart

AttrID = 5_{10}. Es un reporte resumido de la transacción. El formato de este Atributo se encuentra definido en [ISO - EFC]. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**SessionTime**	01.01.1990, 00:00:00 a 31.12.2117, 23:59:58	tttt tttt	Tipo DateAndTime, formato definido en [ISO – EFC]. Resolución de 2 segundos.
2	Instante en que se ejecuta la sesión, con una resolución de 2 segundos.		tttt tttt	
3			tttt tttt	
4			tttt tttt	
5	**SessionServiceProvider**		0111 0010	Formato idéntico al de ContractProvider. En un sistema con múltiples operadores, SessionServiceProvider puede ser diferente a ContractProvider.
6	Concesión que provee el servicio de la sesión.		01CC CCCC	
7			CCCC CCCC	
8	**StationLocation**	Definido por el operador	SSSS SSSS	Definiciones deben registrarse en el MOPTT
9	Código con el que se identifica el punto de cobro o plaza		SSSS SSSS	
10			SSSS	
	SessionLocation	Definido por el operador	LLLL	Definiciones deben registrarse en el MOPTT
11	Pista dentro de punto de cobro o plaza		LLLL	
	TypeOfSession Designa el tipo de servicio	Especificado en [ISO – EFC]	TTTT	
12	**SessionResultOperational** Código para designar si una sesión ha sido completada con éxito o no con relación a aspectos operacionales	0: sesión exitosa 10_{hex}: sesión no exitosa	RRRR RRRR	Otros valores deben ser acordados con el MOPTT
13	**SessionResultFinancial** Código para designar si una sesión ha sido completada con éxito o no con relación a aspectos financieros	0: sesión exitosa 10_{hex}: sesión no exitosa	RRRR RRRR	Otros valores deben ser acordados con el MOPTT

Atributo: SessionClass
AttrID = 6_{10}. Entrega la clase aplicada en la sesión con fines del cálculo de la tarifa, y además la clase declarada en el transponder. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	**SessionTariffClass** Clase específica aplicada durante la sesión	0 a 255_{10}	TTTT TTTT	Permite reproducir el cálculo de la tarifa aplicada.
2	**SessionClaimedClass** Clase declarada en el transponder	0 a 255_{10}	CCCC CCCC	

Atributo: ReceiptAuthenticator
AttrID = 13_{10}. Autenticador evaluado sobre ReceiptServicePart y SessionClass. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		4	0000 0100	Número de bytes del autenticador = 4_{10}
2		0 a $4.294.967.295_{10}$	AAAA AAAA	Similar al Atributo ReceiptAuthenticator definido en [A1], excepto que es evaluado sobre ReceiptServicePart y SessionClass.
3			AAAA AAAA	
4			AAAA AAAA	
5			AAAA AAAA	

Atributo: VehicleClass

AttrID = 17_{10}. Clase del vehículo declarada en el transponder. Acceso: Sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		Clases en Concesiones Urbanas: 00_2: Auto y Camioneta, con o sin acoplado 01_2: Bus, Camión sin acoplado 10_2: Camión con acoplado 11_2: Motocicleta	**0**cc	Para simplificar la interoperabilidad, se recomienda que ambas clases, urbanas e interurbanas, sean grabadas en VehicleClass. Resulta entonces: Motocicleta: 0**11**0 0**000**$_2$ = 60_{16} Auto y Camioneta: 0**00**0 0**001**$_2$ = 01_{16} Auto y Camioneta con acoplado: 0**00**0 0**010**$_2$ = 02_{16} Bus de 2 ejes: 0**01**0 0**011**$_2$ = 23_{16} Camión de 2 ejes: 0**01**0 0**100**$_2$ = 24_{16} Bus con más de 2 ejes: 0**01**0 0**101**$_2$ = 25_{16} Camión con más de 2 ejes sin acoplado: 0**01**0 0**110**$_2$ = 26_{16} Camión con acoplado: 0**10**0 0**110**$_2$ = 46_{16}
		Clases en Concesiones Interurbanas: 000_2: Motocicleta 001_2: Auto y Camioneta 010_2: Auto y Camioneta con acoplado 011_2: Bus de 2 ejes 100_2: Camión de 2 ejes 101_2: Bus con más de 2 ejes 110_2: Camión con más de 2 ejes	**0 0**ccc	

Atributo: EquipmentStatus

AttrID = 26_{10}. Información relativa a la aplicación de cobro de peaje o EFC, referente al status del equipamiento. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1	BlackList		X	0 = OK, 1 = en lista negra
	GrayList		X	0 = OK, 1 = en lista gris
	YellowList		X	0 = OK, 1 = en lista amarilla
	GreenList		X	0 = OK, 1 = ver sección 7.4.4, Lista Verde
	OBETransactionCounter		CCCC	Contador de transacciones del Transponder
2			CCCC CCCC	

Atributo: Spare

AttrID = 98_{10}. Reservado para uso futuro. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1				Definición y uso a ser acordados con el MOPTT
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				

Claves de Seguridad

Acceso: Sin acceso.

AttrId	Clave	Valor	Número de Bytes	Comentarios
111_{10}	**ElementAuthenticationKeyA1**	Clave DES o 3-DES. Si es una clave 3-DES, la mitad izquierda debe ser igual a la mitad derecha, para mantener la compatibilidad con [A1].	8/16	Claves de autenticación en el dominio del ContractProvider (emisor del transponder)
112_{10}	**ElementAuthenticationKeyA2**		8/16	
113_{10}	**ElementAuthenticationKeyF1**		8/16	Claves de autenticación en el dominio del MOPTT
114_{10}	**ElementAuthenticationKeyF1**		8/16	
115_{10}	**ElementAuthenticationKeyI1**		8/16	Claves interoperables de autenticación, generadas por el MOPTT y conocidas por todas las concesiones
116_{10}	**ElementAuthenticationKeyI2**		8/16	
117_{10}	**ElementAuthenticationKeyI3**		8/16	
118_{10}	**ElementAuthenticationKeyI4**		8/16	
120_{10}	**ElementAccessKey**		8/16	Clave interoperable de acceso, generada por el MOPTT y conocida por todas las concesiones

B.3 ATRIBUTOS DEL ELEMENTO DEL EMISOR DEL TRANSPONDER

Atributo: EFC-ContextMark
AttrID = 0_{10}. Denota el contexto específico de cobro de peaje. Incluye la concesión emisora del transponder, el tipo de contrato y la versión de éste. El formato de este Atributo se encuentra definido en [ISO - EFC]. Acceso: no está permitido el acceso directo, este Atributo se transmite en la VST.

La estructura de este Atributo es idéntica a la del EFC-ContextMark del Elemento interoperable especificado en la sección B.2. Rigen las mismas definiciones allí indicadas para los valores de los datos elementales de este EFC-ContextMark.

El MOPTT llevará en el documento complementario [MOPTT-ST1-1] un registro actualizado de los valores vigentes de este EFC-ContextMark, y cuidará que en el dominio de cada concesión no existan valores de EFC-ContextMark duplicados, ni coincidentes con los ApplicationContextMark de otras aplicaciones. El documento [MOPTT-ST1-1], en su última versión, se distribuirá a las concesiones cada vez que sea modificado.

Atributo: Scratchpad
AttrID = 96_{10}. Reservado para uso futuro. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1				Definición y uso a ser acordados con el MOPTT
2				
3				
4				
5				
6				

Claves de Seguridad
Acceso: Sin acceso.

AttrId	Clave	Valor	Número de Bytes	Comentarios
120_{10}	**ElementAccessKey**	Clave DES o 3-DES. Si es una clave 3-DES, la mitad izquierda debe ser igual a la mitad derecha, para mantener la compatibilidad con [A1].	8/16	Clave de acceso privada, generada por el emisor del transponder y conocida solamente por él.

B.4 ATRIBUTOS DEL ELEMENTO PARA GESTIÓN DE ESTACIONAMIENTOS

Atributo: PM-ContextMark
AttrID = 0_{10}. Denota el contexto específico de gestión de estacionamientos. Incluye la concesión emisora del transponder, el tipo de contrato y la versión de éste. El formato de este Atributo se encuentra definido en [ISO - EFC]. Acceso: no está permitido el acceso directo, este Atributo se transmite en la VST.

La estructura de este Atributo es idéntica a la del EFC-ContextMark del Elemento interoperable especificado en la sección B.2. Rigen las mismas definiciones allí indicadas para los valores de los datos elementales de PM-ContextMark.

El MOPTT llevará en el documento complementario [MOPTT-ST1-1] un registro actualizado de los valores vigentes de PM-ContextMark, y cuidará que en el dominio de cada concesión no existan valores de PM-ContextMark duplicados, ni coincidentes con los ApplicationContextMark de otras aplicaciones. El documento [MOPTT-ST1-1], en su última versión, se distribuirá a las concesiones cada vez que sea modificado.

Atributo: ContractSerialNumber
AttrID = 1_{10}. Número de serie que designa el contrato individual con el usuario. Su valor es asignado a discreción por el "ContractProvider". El formato de este Atributo está definido en [ISO - EFC]. Acceso: Sólo lectura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 4.294.967.295 Definido por el emisor del transponder	SSSS SSSS	La lista de clientes, incluido el respectivo "ContractSerialNumber", debe registrarse en el MOPTT.
2			SSSS SSSS	
3			SSSS SSSS	
4			SSSS SSSS	

Claves de Seguridad
Acceso: Sin acceso.

AttrID	Clave	Valor	Número de Bytes	Comentarios
111_{10}	**ElementAuthenticationKey**	Clave DES o 3-DES. Si es una clave 3-DES, la mitad izquierda debe ser igual a la mitad derecha, para mantener la compatibilidad con [A1].	8/16	Clave interoperable de autenticación, generada por el MOPTT y conocida por todas las concesiones
120_{10}	**ElementAccessKey**		8/16	Clave interoperable de acceso, generada por el MOPTT y conocida por todas las concesiones

B.5 ATRIBUTOS DEL ELEMENTO DE SONDA DE TRÁFICO

Atributo: Private-ContextMark
AttrID = 0_{10}. Denota el contexto específico privado. Incluye la concesión emisora del transponder, el tipo de contrato y la versión de éste. El formato de este Atributo se encuentra definido en [ISO - EFC]. Acceso: no está permitido el acceso directo, este Atributo se transmite en la VST.

La estructura de este Atributo es idéntica a la del EFC-ContextMark del Elemento interoperable especificado en la sección B.2. Rigen las mismas definiciones allí indicadas para los valores de los datos elementales de Private-ContextMark.

El MOPTT llevará en el documento complementario [MOPTT-ST1-1] un registro actualizado de los valores vigentes de Private-ContextMark, y cuidará que en el dominio de cada concesión no existan valores de Private-ContextMark duplicados, ni coincidentes con los ApplicationContextMark de otras aplicaciones. El documento [MOPTT-ST1-1], en su última versión, se distribuirá a las concesiones cada vez que sea modificado.

Atributo: TemporaryID
AttrID = 96_{10}. Identificación transitoria RSE. Acceso: Lectura / Escritura.

Byte #	Dato Elemental	Valor	Bits b_7 b_0	Comentarios
1		0 a 16.777.215	RRRR RRRR	Valor aleatorio
2			RRRR RRRR	
3			RRRR RRRR	

Security Key
Acceso: Sin acceso.

AttrId	Clave	Valor	Número de Bytes	Comentarios
120_{10}	**ElementAccessKey**	Clave DES o 3-DES. Si es una clave 3-DES, la mitad izquierda debe ser igual a la mitad derecha, para mantener la compatibilidad con [A1].	8/16	Clave interoperable de acceso, generada por el MOPTT y conocida por todas las concesiones

Anexo C. Codificación de las Transacciones

C.1 INICIALIZACIÓN

C.1.1 BEACON SERVICE TABLE (BST)

BST es similar para todas las aplicaciones consideradas, solamente el byte # 18 es dependiente de la aplicación.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Broadcast LID		1111 111**1**	Direccionamiento del enlace: modo broadcast
3	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Valor irrelevante. Colocar en 0.
	MAC control field. reserved bits		000	Bits reservados
4	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	
	LLC control field. M		00	
	LLC control field. reserved bits		11	Bits no usados. Mantenerlos en 1.
5	Fragmentation header		**1001 1**001	Sin fragmentación. Número **PDU**, usar valor igual a 0011_2. Nunca usar 0000_2 o 0001_2.
6	BST	SEQUENCE	1000	INITIALISATION.request
	{			
	Indicador de Opción		0	nonmandApplictions no presentes
	BeaconId.ManufacturerId	INTEGER (0..65535)	mmm	(MSB) Ver lista de fabricantes formalizada según ISO 14816 y administradas por NNI.
7			mmmm mmmm	
8			mmmm m	
	BeaconId.IndividualId	INTEGER (0..2^{27}-1)	iii	(MSB) 27 bits para identificación del fabricante del interrogador
9			iiii iiii	
10			iiii iiii	
11			iiii iiii	
12	Time	TimeReal	tttt tttt	(MSB) 32 bits representan tiempo real en formato UNIX.
13			tttt tttt	
14			tttt tttt	
15			tttt tttt	
16	Profile	INTEGER (0..127,...)	**0**ppp pppp	Sin extensión, Perfil p $p=0_{10}$: subportadora de 1,5 MHz $p=1_{10}$: subportadora de 2,0 MHz
17	mandApplications	SEQUENCE (0..127,..) OF	**0**000 0001	Sin extensión, cantidad de mandApplications = 1
	{			

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
18	Indicador de Opción		0	EID no presente
	Indicador de Opción		0	Parámetro no presente
	AID	DSRCApplicationEntityID	**0**0 0001	Sin extensión. AID = 1_{10} (EFC o peaje), 6_{10} (Estacionamiento) o 29_{10} (Sonda de Tráfico). Codificación mostrada para EFC.
	}			
19	ProfileList	SEQUENCE (0..127,..) OF	**0**000 0000	Sin extensión, número de perfiles en lista = 0
	}	Profiles		
20	FCS		xxxx xxxx	Secuencia verificadora de la trama
21			xxxx xxxx	
22	FLAG		0111 1110	"Flag" de término

C.1.2 SOLICITUD DE VENTANA DE COMUNICACIONES PRIVADA (PrWRq)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Dirección privada de un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	0	La trama no contiene un LPDU
	MAC control field. D	1	Dirección Es "Up Link"
	MAC control field. R	1	Se solicita ventana privada de "Up link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. Reserved bits	0000	Bits reservados
7	FCS	xxxx xxxx	Secuencia verificadora de la trama
8		xxxx xxxx	
9	FLAG	0111 1110	"Flag" de término

C.1.3 ASIGNACIÓN DE VENTANA DE COMUNICACIONES PRIVADA (PrWA)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace de un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	0	La trama no contiene un LPDU
	MAC control field. D	0	Dirección es "down link"
	MAC control field. R	1	RSE asigna ventana privada en el "Up link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. S	S	Bit de Secuencia
	MAC control field. Reserved bits	000	Bits reservados
7	FCS	xxxx xxxx	Secuencia verificadora de la trama

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
8		xxxx xxxx	
9	FLAG	0111 1110	"Flag" de término

C.1.4 VEHICLE SERVICE TABLE (VST) EN TRANSACCIÓN NACIONAL DE PEAJE INTEROPERABLE

Los Bytes 32 a 51 reflejan la presencia del Elemento del emisor del transponder.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace de un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. Reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	
	LLC control field. M		00	
	LLC control field. Reserved bits		11	Bits no usados. Mantenerlos en 1.
8	Fragmentation header		**1001 1**001	Sin fragmentación. Valor de número **PDU** igual al del BST.
9	VST	SEQUENCE	1001	INITIALISATION.response
	{			
	Fill	BIT STRING (SIZE(4))	0000	Rellenar con 0
10	Profile	INTEGER (0..127,...)	**0**ppp pppp	Sin extensión, perfil p
11	Applications	SEQUENCE (0..127,...) OF	**0**000 0010	Sin extensión, 2 aplicaciones
12	**{**			Inicio de primera aplicación
	Indicador de Opción		1	EID presente
	Indicador de Opción		1	Parámetro presente
	AID	DSRCApplicationEntityID	**0**0 0001	Sin extensión, AID = 1 (EFC)
13	EID		eeee eeee	EID del Elemento de peaje interoperable al interior del OBE y relativo a una ContextMark.
14	Parameter	CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
15			**0**001 0000	Sin extensión, largo de OCTET STRING = 16_{10}
16	EFC-ContextMark	SEQUENCE		
	{			
	ContractProvider	SEQUENCE		
	{			
	CountryCode	BIT STRING (SIZE(10))	0111 0010	(MSB) 10 bits para código del país según ISO
17			01	3166 con codificación binaria ITA2 basada en ISO 14816. Ver Anexo B. Valor: 01110 01001_2 = 457_{10} para Chile

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	IssuerIdentifier	INTEGER (0..16383)	dd dddd	(MSB) 14 bits para identificar al emisor del transponder (ver Anexo B)
18			dddd dddd	
	}			
19	TypeOfContract	OCTET STRING (SIZE(2))	tttt tttt	(MSB) Tipo t de contrato (ver Anexo B)
20			tttt tttt	
21	ContextVersion	INTEGER (0..127,..)	**0**vvv vvvv	Sin extensión, ContextVersion v (ver Anexo B)
	}			
22		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
23			**0**000 0010	Sin extensión, largo de OCTET STRING = 2_{10}
24	OBEGroupID	INTEGER (0..65535)	gggg gggg	Identificador del grupo a que pertenece el OBE
25			gggg gggg	
26		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
27			**0**000 0100	Sin extensión, largo de OCTET STRING = 4_{10}
28	RndOBE	INTEGER (0..2^{32}-1)	rrrr rrrr	Número aleatorio usado junto con EAcKey para calcular AC_CR
29			rrrr rrrr	
30			rrrr rrrr	
31			rrrr rrrr	
	}			Fin de primera aplicación
32	{			Inicio de segunda aplicación
	Indicador de Opción		1	EID presente
	Indicador de Opción		1	Parámetro presente
	AID	DSRCApplicationEntityID	**0**0 0001	Sin extensión, AID = 1 (EFC)
33	EID		eeee eeee	EID del Elemento del emisor del transponder al interior del OBE, relativo a una ContextMark
34	Parameter	CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
35			**0**001 0000	Sin extensión, largo de OCTET STRING = 16_{10}
36	EFC-ContextMark	SEQUENCE		
	{			
	ContractProvider	SEQUENCE		
	{			
	CountryCode	BIT STRING (SIZE(10))	0111 0010	(MSB) 10 bits para código del país según ISO 3166 con codificación binaria ITA2 basada en ISO 14816. Ver Anexo B. Valor: 01110 01001$_2$ = 457$_{10}$ para Chile
37			01	
	IssuerIdentifier	INTEGER (0..16383)	dd dddd	(MSB) 14 bits para identificar al emisor del transponder (ver Anexo B)
38			dddd dddd	
	}			
39	TypeOfContract	OCTET STRING (SIZE(2))	tttt tttt	(MSB) Tipo t de contrato (ver Anexo B)
40			tttt tttt	
41	ContextVersion	INTEGER (0..127,..)	**0**vvv vvvv	Sin extensión, ContextVersion v (ver Anexo B)
	}			
42		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
43			**0**000 0010	Sin extensión, largo de OCTET STRING = 2_{10}
44	OBEGroupID	INTEGER (0..65535)	gggg gggg	Identificador del grupo a que pertenece el OBE
45			gggg gggg	
46		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
47			**0**000 0100	Sin extensión, largo de OCTET STRING = 4_{10}

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
48	RndOBE	INTEGER (0..2^{32}-1)	rrrr rrrr	Número aleatorio usado junto con EAcKey para calcular AC_CR
49			rrrr rrrr	
50			rrrr rrrr	
51			rrrr rrrr	
	}			Fin de segunda aplicación
52	ObeConfiguration	SEQUENCE		
	{			
	Indicador de Opción		1	ObeStatus presente
	EquipmentClass	INTEGER (0..32767)	xxx xxxx	(MSB) Ver Anexo B
53			xxxx xxxx	
54	ManufacturerId	INTEGER (0..65535)	mmmm mmmm	(MSB) Ver Anexo B
55			mmmm mmmm	1: Kapsch, 2: Alcatel, 3: Combitech, 4: CSSI 6: QFree
56	obeStatus	SEQUENCE		
	{			
	StatusFlags	BIT STRING (SIZE(16))	A	Ausencia de tarjeta inteligente (si es aplicable)
			I	ICC no reconocida (si es aplicable)
			B	Falla de batería (si es aplicable)
			P	Error en interfaz de periférico (si es aplicable)
			T	Manipulación ilegal del OBE
			xxx	Último estado antes de pasar a reposo: (si es aplicable) 000: BLOCKED 001: WAIT 010: INIT 011: READY 100: DATA
57			ssss sss	Reservados para uso privado
			x	Transponder fue desmontado del vehículo
	} } }			
58	FCS		xxxx xxxx	Secuencia verificadora de la trama
59			xxxx xxxx	
60	FLAG		0111 1110	"Flag" de término

C.1.5 VEHICLE SERVICE TABLE (VST) EN TRANSACCIÓN DE GESTIÓN DE ESTACIONAMIENTOS

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	0	LPDU tipo comando

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	MAC control field. Reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	
	LLC control field. M		00	
	LLC control field. Reserved bits		11	Bits no usados. Mantenerlos en 1.
8	Fragmentation header		1001 1001	Sin fragmentación. Valor de número **PDU** igual al del BST.
9	VST	SEQUENCE	1001	INITIALISATION.response
	{			
	Fill	BIT STRING (SIZE(4))	0000	Rellenar con 0
10	Profile	INTEGER (0..127,...)	0ppp pppp	Sin extensión, perfil p
11	Applications	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, 1 aplicación
12	{			Inicio de aplicación
	Indicador de Opción		1	EID presente
	Indicador de Opción		1	Parámetro presente
	AID	DSRCApplicationEntityID	00 0110	Sin extensión, AID = 6_{10} (Estacionamientos)
13	EID		eeee eeee	EID del Elemento de G. de Estacionamientos al interior del OBE, ligado a una ContextMark
14	Parameter	CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
15			0001 0000	Sin extensión, largo de OCTET STRING = 16_{10}
16	PM-ContextMark	SEQUENCE		
	{			
	ContractProvider	SEQUENCE		
	{			
	CountryCode	BIT STRING (SIZE(10))	0111 0010	(MSB) 10 bits para código del país según ISO
17			01	3166 con codificación binaria ITA2 basada en ISO 14816. Ver Anexo B. Valor: 01110 01001$_2$ = 457$_{10}$ para Chile
	IssuerIdentifier	INTEGER (0..16383)	dd dddd	(MSB) 14 bits para identificar al emisor del transponder (ver Anexo B)
18			dddd dddd	
	}			
19	TypeOfContract	OCTET STRING (SIZE(2))	tttt tttt	(MSB) Tipo t de contrato (ver Anexo B)
20			tttt tttt	
21	ContextVersion	INTEGER (0..127,..)	0vvv vvvv	Sin extensión, ContextVersion v (ver Anexo B)
	}			
22		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
23			0000 0010	Sin extensión, largo de OCTET STRING = 2_{10}
24	OBEGroupID	INTEGER (0..65535)	gggg gggg	Identificador del grupo a que pertenece el OBE
25			gggg gggg	
26		CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
27			0000 0100	Sin extensión, largo de OCTET STRING = 4_{10}
28	RndOBE	INTEGER (0..2^{32}-1)	rrrr rrrr	Número aleatorio usado junto con EAcKey para calcular AC_CR
29			rrrr rrrr	
30			rrrr rrrr	
31			rrrr rrrr	
	}			Fin de la aplicación

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
32	ObeConfiguration	SEQUENCE		
	{			
	Indicador de Opción		1	ObeStatus presente
	EquipmentClass	INTEGER (0..32767)	xxx xxxx	(MSB) Ver Anexo B
33			xxxx xxxx	
34	ManufacturerId	INTEGER (0..65535)	mmmm mmmm	(MSB) Ver Anexo B
35			mmmm mmmm	1: Kapsch, 2: Alcatel, 3: Combitech, 4: CSSI 6: QFree
36	obeStatus	SEQUENCE		
	{			
	StatusFlags	BIT STRING (SIZE(16))	A	Ausencia de tarjeta inteligente (si es aplicable)
			I	ICC no reconocida (si es aplicable)
			B	Falla de batería (si es aplicable)
			P	Error en interfaz de periférico (si es aplicable)
			T	Manipulación ilegal del OBE
			xxx	Último estado antes de pasar a reposo: (si es aplicable) 000: BLOCKED 001: WAIT 010: INIT 011: READY 100: DATA
37			ssss sss	Reservados para uso privado
			x	Transponder fue desmontado del vehículo
	} } }			
38	FCS		xxxx xxxx	Secuencia verificadora de la trama
39			xxxx xxxx	
40	FLAG		0111 1110	"Flag" de término

C.1.6 VEHICLE SERVICE TABLE (VST) EN TRANSACCIÓN DE SONDA DE TRÁFICO

La VST para la aplicación de Sonda de Tráfico es idéntica a la de la Gestión de Estacionamientos, excepto en los bytes 12 a 16:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
12	{			Inicio de la aplicación
	Indicador de Opción		1	EID presente
	Indicador de Opción		1	Parámetro presente
	AID	DSRCApplicationEntityID	**0**1 1101	Sin extensión, AID = 29_{10} (Sonda de Tráfico)
13	EID		eeee eeee	EID del Elemento de Sonda de Tráfico dentro del OBE y relacionado con una ContextMark
14	Parameter	CONTAINER	0000 0010	CHOICE 2_{10} = OCTET STRING
15			**0**001 0000	Sin extensión, largo de OCTET STRING = 16_{10}
16	Private-ContextMark	SEQUENCE		

C.2 NÚCLEO DE LA TRANSACCIÓN PARA PEAJE INTEROPERABLE

C.2.1 PRESENTACIÓN: SERVICIOS CONCATENADOS: GET_STAMPED.REQUEST, GET.REQUEST Y Y GET_NONCE.REQUEST OPCIONAL (ACn)

Existen variaciones en la codificación, dependientes de los casos específicos definidos en el Anexo A, sección A.2.2.2. Además, algunas porciones son opcionales:

- Byte 39, presente sólo cuando el operador utiliza el Atributo “Receipt Authenticator”.
- Byte 41, presente sólo cuando el operador utiliza el Atributo de reserva “Spare”.
- Bytes 42 a 45, presentes sólo en la configuración 2 de AIAs, si en un comando posterior se va a acceder al elemento de sistema.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	“Flag” inicial
2	Private LID		xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx0	
4			xxxx xxx0	
5			xxxx xxx1	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es “Down Link”
	MAC control field. A		1	RSE asigna ventana privada en el “Up Link”
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	Comando ACn
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	Bits no usados. Mantenerlos en 1.
8	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Primer servicio concatenado.
9			0000	ACTION.request
	GET_STAMPED.request	SEQUENCE		
	{			
	Indicador de Opción		1	Credencial de Acceso presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Se espera respuesta
10	EID	INTEGER(0..127,...)	**0**eee eeee	EID de Elemento de Peaje Interoperable en el OBE, relacionado con una ContextMark.
11	ActionType	INTEGER(0..127,...)	**0**000 0000	Sin extensión, GET_STAMPED.request = 0
12	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, largo = 4_{10} bytes
	{			
13	AC_CR		aaaa aaaa	Credencial de Acceso calculada por RSE usando RndOBE y la clave de acceso al
14			aaaa aaaa	

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
15			aaaa aaaa	Elemento EAcKey.
16			aaaa aaaa	
	}			
17	ActionParameter	CONTAINER	**0**001 0001	Sin extensión, CHOICE 17_{10} = GetStampedRq
	{			
	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
18	AttributeId		**0**000 0001	Sin extensión, número de AttributeIDs = 1
	{			
19	EquipmentStatus ID		0001 1010	EquipmentStatus ID = 26_{10}
	} }			
20	nonce	OCTET STRING	**0**000 0100	Sin extensión, largo de RndRSE = 4_{10} bytes
	{			
21	RndRSE		rrrr rrrr	Valor aleatorio entregado por el RSE, necesario para calcular OBEAuthenticator
22			rrrr rrrr	
23			rrrr rrrr	
24			rrrr rrrr	
	}			
25	KeyRef		**0**11y yyyy	Referencia a clave ElementAuthenticationKey usada en el cálculo del OBEAuthenticator: Caso 1A: EAuK_A (111_{10} ... 112_{10}). Casos 1B y 2: EAuK_I (115_{10} ... 118_{10}).
	} }			
26	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff**: valor de **número PDU** incrementado secuencialmente. 2º servicio concatenado.
27	GET.request	SEQUENCE	0110	GET.request
	{			
	Indicador de Opción		1	Credencial de Acceso presente
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeIdList presente
	Fill	BIT STRING(SIZE(1))	0	Colocar en 0
28	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento de peaje Interoperable
29	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, largo de AC_CR = 4_{10} bytes
	{			
30	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando RndOBE y el ElementAccessKey EAcKey del Elemento de peaje Interoperable
31			aaaa aaaa	
32			aaaa aaaa	
33			aaaa aaaa	
	}			
34	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
	AttributeId		**0**000 01xx	Sin extensión, cantidad de AttributeIDs = 5_{10}, 6_{10} o 7_{10}, según se lean o no los atributos "ReceiptAuthenticator" y "Spare"
	{			
35	ContractSerialNumber		0000 0001	attributeID = 1_{10} (ContractSerialNumber)
36	ContractValidity		0000 0010	attributeID = 2_{10} (ContractValidity)
37	ReceiptServicePart		0000 0101	attributeID = 5_{10} (ReceiptServicePart)
38	SessionClass		0000 0110	attributeID = 6_{10} (SessionClass)
39	ReceiptAuthenticator		0000 1101	attributeID = 13_{10} (ReceiptAuthenticator)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
				Opcional
40	VehicleClass		0001 0001	attributeID = 17_{10} (VehicleClass)
41	Spare		0110 0010	attributeID = 98_{10} (Spare) Opcional
	} } }			
42	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Tercer servicio concatenado (opcional para configuración 2 de AIAs).
43	GET-NONCE.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		0	Credencial de acceso no presente
	Indicador de Opción		0	ActionParameter no presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
44	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento del emisor del transponder si más adelante se va a acceder a dicho elemento, y EID = 0 si más adelante se va a acceder al Elemento de sistema
45	ActionType	INTEGER(0..127,...)	**0**000 0110	Sin extensión, GET-NONCE.request = 6_{10}
	}			
46	FCS		xxxx xxxx	Secuencia verificadora de la trama
47			xxxx xxxx	
48	FLAG		0111 1110	"Flag" de término

C.2.2 PRESENTACIÓN: SERVICIOS CONCATENADOS: GET_STAMPED.RESPONSE, GET.RESPONSE Y GET_NONCE.RESPONSE OPCIONAL (ACn)

Las siguientes porciones son dependientes de los comandos recibidos:

- Bytes 60 a 66 sólo están presentes cuando se solicita el valor del Atributo "ReceiptAuthenticator".
- Bytes 70 a 85 sólo están presentes cuando se solicita el valor del Atributo "Spare".
- Bytes 86 a 95 sólo están presentes cuando se solicita el valor "nonce".

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	1	LPDU tipo respuesta
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. n	n	Bit n del comando ACn
	LLC control field. M	11	

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	LLC control field. P/F		1	Bit final = 1
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR		0000	Respuesta disponible
	LLC status field. CCCC		0000	Comando aceptado
9	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con 1er servicio concatenado: GET_STAMPED.request.
10			0001	ACTION.response
	GET_STAMPED.response	SEQUENCE		
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	Parámetro de respuesta presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con valor 0
11	EID	INTEGER (0..127,...)	0eee eeee	Sin extensión, EID de elemento de peaje Interoperable
12	ResponseParameter	CONTAINER	0001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{			
13	AttributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, cantidad de atributos = 1
	{			
	Attributes	SEQUENCE		
	{			
14	AttributeID		0001 1010	EquipmentStatus ID = 26_{10}
15	AttributeValue	CONTAINER	0011 1010	CHOICE: 58_{10} = EquipmentStatus
	{			
16	EquipmentStatus		ssss ssss	Valor de EquipmentStatus
17			ssss ssss	
	} } }			
18	Authenticator	OCTET STRING	0000 0100	Sin extensión. Longitud del “string“= 4 bytes
	{			
19	OBEAuthenticator		xxxx xxxx	Autenticador calculado sobre EquipmentStatus, usando la clave ElementAuthenticationKey seleccionada por keyRef, y el número aleatorio RndRSE
20			xxxx xxxx	
21			xxxx xxxx	
22			xxxx xxxx	
	} } }			
23	Fragmentation header		1fff f001	Sin fragmentación. **ffff**: mismo valor de número PDU recibido con 2º servicio concatenado: GET.request.
24	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
25	EID	INTEGER(0..127,...)	0eee eeee	Sin extensión, EID de elemento de peaje Interoperable.
26	AttributeList	SEQUENCE (0..127,...) OF	0000 01xx	Sin extensión, el número de atributos en la lista es 5_{10}, 6_{10} o 7_{10}, dependiendo de si se leen o no los atributos ReceiptAuthenticator y Spare.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			
27	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0000 0001	AttributeId = 1_{10} (ContractSerialNumber)
28	Attribute Value	CONTAINER	0010 0001	CONTAINER CHOICE = 33_{10}
	{			
29	ContractSerialNumber		aaaa aaaa	Valor de ContractSerialNumber
30			aaaa aaaa	
31			aaaa aaaa	
32			aaaa aaaa	
	}			
33	AttributeId	INTEGER(0..127,...)	0000 0010	AttributeId = 2_{10} (ContractValidity)
34	Attribute Value	CONTAINER	0010 0010	CONTAINER CHOICE = 34_{10}
	{			
35	ContractValidity		xxxx xxxx	Valor de ContractValidity.ContractRestrictions
36			xxxx xxxx	
37			xxxx xxxx	
38			xxxx xxxx	
39			xxxx xxxx	Valor de ContractValidity.ContractExpiryDate
40			xxxx xxxx	
	}			
41	AttributeId	INTEGER(0..127,...)	0000 0101	AttributeId = 5_{10} (ReceiptServicePart)
42	Attribute Value	CONTAINER	0010 0101	CONTAINER CHOICE = 37_{10}
	{			
43	ReceiptServicePart		xxxx xxxx	Valor de ReceiptServicePart.SessionTime
44			xxxx xxxx	
45			xxxx xxxx	
46			xxxx xxxx	
47			xxxx xxxx	Valor de ReceiptServicePart.Provider
48			xxxx xxxx	
49			xxxx xxxx	
50			xxxx xxxx	Valor de ReceiptServicePart.StationLocation
51			xxxx xxxx	
52			xxxx	
			xxxx	Valor de ReceiptServicePart.SessionLocation
53			xxxx	
			xxxx	Valor de ReceiptServicePart.TypeOfSession
54			xxxx xxxx	ReceiptServicePart.SessionResultOperational
55			xxxx xxxx	ReceiptServicePart.SessionResultFinancial
	}			
56	AttributeId	INTEGER(0..127,...)	0000 0110	AttributeId = 6_{10} (SessionClass)
57	Attribute Value	CONTAINER	0010 0110	CONTAINER CHOICE = 38_{10}
	{			
58	SessionClass		xxxx xxxx	Valor de SessionClass.SessionTariffClass
59			xxxx xxxx	Valor de SessionClass.SessionClaimedClass
	}			
60	AttributeId	INTEGER(0..127,...)	0000 1101	AttributeId = 13_{10} (ReceiptAuthenticator)
61	Attribute Value	CONTAINER	0010 1101	CONTAINER CHOICE = 45_{10}
	{			

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
62	ReceiptAuthenticator		0000 0100	Largo de ReceiptAuthenticator = 4_{10} bytes
63			aaaa aaaa	Valor de ReceiptAuthenticator
64			aaaa aaaa	
65			aaaa aaaa	
66			aaaa aaaa	
	}			
67	AttributeId	INTEGER(0..127,...)	0001 0001	AttributeId = 17_{10} (VehicleClass)
68	Attribute Value	CONTAINER	0011 0001	CONTAINER CHOICE = 49_{10}
	{			
69	VehicleClass		xxxx xxxx	Valor de clase declarada del vehículo
	}			
70	AttributeId	INTEGER(0..127,...)	0110 0010	AttributeId = 98_{10} (Spare)
71	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10}
72	{		0000 1101	Longitud de Spare = 13_{10} bytes
73	Spare		xxxx xxxx	Valor de Spare
74			xxxx xxxx	
75			xxxx xxxx	
76			xxxx xxxx	
77			xxxx xxxx	
78			xxxx xxxx	
79			xxxx xxxx	
80			xxxx xxxx	
81			xxxx xxxx	
82			xxxx xxxx	
83			xxxx xxxx	
84			xxxx xxxx	
85			xxxx xxxx	
	}			
	} } }			
86	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
87	ACTION.response	SEQUENCE	0001	GET_NONCE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		1	ReturnStatus presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
88	EID	INTEGER (0..127,...)	0000 0000	Sin extensión, EID de Elemento de sistema =0
89	Attribute Value		0000 0010	CONTAINER TYPE = 2_{10} (Octet string)
90	{		0000 0100	Sin extensión , largo de nonce = 4 bytes
91	Nonce		nnnn nnnn	Valor nonce
92			nnnn nnnn	
93			nnnn nnnn	
94			nnnn nnnn	
	}			
95			rrrr rrrr	Status
	}			
96	FCS		xxxx xxxx	Secuencia verificadora de la trama
97			xxxx xxxx	

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
98	FLAG	0111 1110	"Flag" de término

C.2.3 PRESENTACIÓN: SERVICIOS CONCATENADOS: GET_STAMPED.RESPONSE, GET.RESPONSE (UI)

Las siguientes porciones son dependientes de los comandos recibidos:

- Bytes 59 a 65 sólo están presentes cuando se solicita el valor del Atributo "ReceiptAuthenticator".
- Bytes 69 a 84 sólo están presentes cuando se solicita el valor del Atributo "Spare".
- Bytes 85 a 94 sólo están presentes cuando se solicita el valor "nonce".

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		1	LPDU tipo respuesta
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		1	No Poll
	LLC control field. M		00	Comando UI
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con primer servicio concatenado: GET_STAMPED.request.
9			0001	ACTION.response
	GET_STAMPED.response	SEQUENCE		
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	Parámetro de respuesta presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con valor 0
10	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID de elemento de peaje Interoperable
11	ResponseParameter	CONTAINER	**0**001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{			
12	AttributeList	SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, cantidad de atributos = 1
	{			
	Attributes	SEQUENCE		
	{			
13	AttributeID		**0**001 1010	EquipmentStatus ID = 26_{10}

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
14	AttributeValue	CONTAINER	0011 1010	CHOICE: 58_{10} = EquipmentStatus
	{			
15	EquipmentStatus		ssss ssss	Valor de EquipmentStatus
16			ssss ssss	
	} } }			
17	Authenticator	OCTET STRING	**0**000 0100	Sin extensión. Longitud del "string"= 4 bytes
	{			
18	OBEAuthenticator		xxxx xxxx	Autenticador calculado sobre EquipmentStatus, usando la clave ElementAuthenticationKey seleccionada por keyRef, y el número aleatorio RndRSE
19			xxxx xxxx	
20			xxxx xxxx	
21			xxxx xxxx	
	} } }			
22	Fragmentation header		1**fff f**001	Sin fragmentación. Mismo valor de **número PDU** recibido con 2º servicio concatenado: GET.request.
23	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
24	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID de elemento de peaje Interoperable.
25	AttributeList	SEQUENCE (0..127,...) OF	**0**000 01xx	Sin extensión, el número de atributos en la lista es 5_{10}, 6_{10} o 7_{10}, dependiendo de si se leen o no los atributos ReceiptAuthenticator y Spare.
	{			
26	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	**0**000 0001	AttributeId = 1_{10} (ContractSerialNumber)
27	Attribute Value	CONTAINER	**0**010 0001	CONTAINER CHOICE = 33_{10}
	{			
28	ContractSerialNumber		aaaa aaaa	Valor de ContractSerialNumber
29			aaaa aaaa	
30			aaaa aaaa	
31			aaaa aaaa	
	}			
32	AttributeId	INTEGER(0..127,...)	**0**000 0010	AttributeId = 2_{10} (ContractValidity)
33	Attribute Value	CONTAINER	**0**010 0010	CONTAINER CHOICE = 34_{10}
	{			
34	ContractValidity		xxxx xxxx	Valor de ContractValidity.ContractRestrictions
35			xxxx xxxx	
36			xxxx xxxx	
37			xxxx xxxx	
38			xxxx xxxx	Valor de ContractValidity.ContractExpiryDate
39			xxxx xxxx	
	}			
40	AttributeId	INTEGER(0..127,...)	**0**000 0101	AttributeId = 5_{10} (ReceiptServicePart)
41	Attribute Value	CONTAINER	**0**010 0101	CONTAINER CHOICE = 37_{10}

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			
42	ReceiptServicePart		xxxx xxxx	Valor de ReceiptServicePart.SessionTime
43			xxxx xxxx	
44			xxxx xxxx	
45			xxxx xxxx	
46			xxxx xxxx	Valor de ReceiptServicePart.Provider
47			xxxx xxxx	
48			xxxx xxxx	
49			xxxx xxxx	Valor de ReceiptServicePart.StationLocation
50			xxxx xxxx	
51			xxxx	
			xxxx	Valor de ReceiptServicePart.SessionLocation
52			xxxx	
			xxxx	Valor de ReceiptServicePart.TypeOfSession
53			xxxx xxxx	ReceiptServicePart.SessionResultOperational
54			xxxx xxxx	ReceiptServicePart.SessionResultFinancial
	}			
55	AttributeId	INTEGER(0..127,...)	0000 0110	AttributeId = 6_{10} (SessionClass)
56	Attribute Value	CONTAINER	0010 0110	CONTAINER CHOICE = 38_{10}
	{			
57	SessionClass		xxxx xxxx	Valor de SessionClass.SessionTariffClass
58			xxxx xxxx	Valor de SessionClass.SessionClaimedClass
	}			
59	AttributeId	INTEGER(0..127,...)	0000 1101	AttributeId = 13_{10} (ReceiptAuthenticator)
60	Attribute Value	CONTAINER	0010 1101	CONTAINER CHOICE = 45_{10}
	{			
61	ReceiptAuthenticator		0000 0100	Largo de ReceiptAuthenticator = 4_{10} bytes
62			aaaa aaaa	Valor de ReceiptAuthenticator
63			aaaa aaaa	
64			aaaa aaaa	
65			aaaa aaaa	
	}			
66	AttributeId	INTEGER(0..127,...)	0001 0001	AttributeId = 17_{10} (VehicleClass)
67	Attribute Value	CONTAINER	0011 0001	CONTAINER CHOICE = 49_{10}
	{			
68	VehicleClass		xxxx xxxx	Valor de clase declarada del vehículo
	}			
69	AttributeId	INTEGER(0..127,...)	0110 0010	AttributeId = 98_{10} (Spare)
70	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10}
71	{		0000 1101	Longitud de Spare = 13_{10} bytes
72	Spare		xxxx xxxx	Valor de Spare
73			xxxx xxxx	
74			xxxx xxxx	
75			xxxx xxxx	
76			xxxx xxxx	
77			xxxx xxxx	
78			xxxx xxxx	
79			xxxx xxxx	
80			xxxx xxxx	

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
81			xxxx xxxx	
82			xxxx xxxx	
83			xxxx xxxx	
84			xxxx xxxx	
	}			
	} } }			
85	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
86	ACTION.response	SEQUENCE	0001	GET_NONCE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		1	ReturnStatus presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
87	EID	INTEGER (0..127,...)	0000 0000	Sin extensión, EID de Elemento de sistema =0
88	Attribute Value		0000 0010	CONTAINER TYPE = 2_{10} (Octet string)
89	{		0000 0100	Sin extensión , largo de nonce = 4 bytes
90	Nonce		nnnn nnnn	Valor Nonce
91			nnnn nnnn	
92			nnnn nnnn	
93			nnnn nnnn	
	}			
94			rrrr rrrr	Status
	}			
95	FCS		xxxx xxxx	Secuencia verificadora de la trama
96			xxxx xxxx	
97	FLAG		0111 1110	"Flag" de término

C.2.4 LLC-STATUS = NE_OK (ACn)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace de un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. n	N	Bit n de comando ACn
	LLC control field. M	11	
	LLC control field. P/F	1	Bit final = 1
	LLC control field. M	01	
	LLC control field. reserved bits	11	No usados, mantenerlos en 1

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
8	LLC status field. RRRR	0011	Respuesta aún no disponible
	LLC status field. CCCC	0000	Comando aceptado
9	FCS	xxxx xxxx	Secuencia verificador de la trama
10		xxxx xxxx	
11	FLAG	0111 1110	"Flag" de término

C.2.5 Autenticación Fiscal y de Contrato en Transacción Nacional de Peaje Interoperable: GET_STAMPED.request, GET.request Opcional (Elemento del Emisor), Ejemplo de GET.request Opcional (Elemento de Sistema) (ACn)

El comando GET_STAMPED.request es idéntico en las autenticaciones Fiscal y de Contrato, excepto por el valor de *key ref* empleado (byte 25). El primer comando GET.request opcional, bytes 26 al 35, sólo existe en el caso de la autenticación Fiscal, cuando se accede al Elemento reservado del emisor del transponder. El segundo comando GET.request opcional, bytes 36 al 46, también existe solamente en el caso de la autenticación Fiscal, cuando se leen los atributos ActivityTimer y BatteryInsertionDate en el Elemento de Sistema.

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx0	
4		xxxx xxx0	
5		xxxx xxx1	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	0	Dirección es "Down Link"
	MAC control field. A	1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. S	S	Bit de secuencia
	MAC control field. reserved bits	000	Bits reservados
7	LLC control field. n	N	Bit n del comando ACn
	LLC control field. M	11	Comando ACn
	LLC control field. P/F	1	1 = Poll, 0 = no Poll
	LLC control field. M	01	
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	Fragmentation header	1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente.
9		0000	ACTION.request
	GET_STAMPED.request SEQUENCE		
	{		
	Indicador de Opción	1	Credencial de acceso presente
	Indicador de Opción	1	ActionParameter presente
	Indicador de Opción	0	IID no presente
	Mode BOOLEAN	1	Se espera respuesta

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
10	EID	INTEGER(0..127,...)	**0**eee eeee	EID de Elemento de peaje Interoperable, relacionado a una ContextMark.
11	ActionType	INTEGER(0..127,...)	**0**000 0000	Sin extensión, GET_STAMPED.request = 0
12	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, largo del "string" = 4_{10} bytes
	{			
13	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando RndOBE y el ElementAccessKey EAcKey.
14			aaaa aaaa	
15			aaaa aaaa	
16			aaaa aaaa	
	}			
17	ActionParameter	CONTAINER	**0**001 0001	Sin extensión, CHOICE 17_{10} = GetStampedRq
	{			
	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
18	AttributeID		**0**000 0001	Sin extensión, cantidad de AttributeIDs = 1
	{			
19	EquipmentStatus ID		0001 1010	EquipmentStatus ID = 26_{10}
	} }			
20	nonce	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{			
21	RndRSE		rrrr rrrr	Número aleatorio entregado por el RSE, necesario para calcular el autenticador Fiscal o del Contrato
22			rrrr rrrr	
23			rrrr rrrr	
24			rrrr rrrr	
	}			
25	KeyRef		**0**11y yyyy	Referencia al ElementAuthenticationKey: Para el cálculo de FiscalAuthenticator: EAuK_F (113_{10} ... 114_{10}). Para el cálculo de ContractAuthenticator: EAuK_A (111_{10} ... 112_{10}).
	} }			
26	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff**: valor de número PDU incrementado secuencialmente. 2º servicio concatenado (Opcional).
27	GET.request	SEQUENCE	0110	GET.request
	{			
	Indicador de Opción		1	Credencial de Acceso presente
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeIdList presente
	Fill	BIT STRING(SIZE(1))	0	Poner en 0
28	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID del elemento del emisor
29	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, largo de AC_CR = 4_{10} bytes
	{			
30	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando el ElementAccessKey EAcKey del Elemento del emisor y: RndOBE, en configuraciones 1, 3 y 4 de AIAs Nonce, en configuración 2 de AIAs
31			aaaa aaaa	
32			aaaa aaaa	
33			aaaa aaaa	
	}			
34	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	AttributeId		**0**000 0001	Sin extensión, número de AttributeIDs = 1
	{			
35	Scratchpad		0110 0000	attributeID = 96_{10} (Scratchpad)
	} } }			
36	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff**: valor de número PDU incrementado secuencialmente. Tercer servicio concatenado (Opcional).
37	GET.request	SEQUENCE	0110	GET.request
	{			
	Indicador de Opción		1	Credencial de Acceso presente
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeIdList presente
	Fill	BIT STRING(SIZE(1))	0	Poner en 0
38	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID = 0 (elemento de sistema)
39	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, largo de AC_CR = 4_{10} bytes
	{			
40	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando el ElementAccessKey EAcKey del Elemento de Sistema y: RndOBE, en configuraciones 1 y 3 de AIAs Nonce, en configuración 2 de AIAs
41			aaaa aaaa	
42			aaaa aaaa	
43			aaaa aaaa	
	}			
44	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
	AttributeId		**0**000 0010	Sin extensión, número de AttributeIDs = 2
	{			
45	ActivityTimer		**0**000 0111	AttributeId de ActivityTimer = 7_{10}, de configuraciones 1, 2 y 3 de AIAs
46	BatteryInsertionDate		**0**001 0000	AttributeId de BatteryInsertionDate = 16_{10}, de configuraciones 1, 2 y 3 de AIAs
	} } }			
47	FCS		xxxx xxxx	Secuencia verificadora de la trama
48			xxxx xxxx	
49	FLAG		0111 1110	"Flag" de término

Para leer los atributos ActivityTimer y BatteryInsertionDate de la configuración 4 de AIAs, se usa la siguiente codificación a partir del byte 36:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
36	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Tercer servicio concatenado (alternativo para configuración 4 de AIAs).
37	PRIVATE.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		0	Credencial de acceso no presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
38	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID del Elemento de sistema
39	ActionType	INTEGER(0..127,...)	**0**111 0111	Sin extensión, PRIVATE.request = 119_{10}
40	Action Parameter	CONTAINER	0000 0010	Sin extensión, Type = 2_{10},(Octet String)
41			0000 0101	Longitud del parámetro = 5_{10}
42			0001 1010	Tipo de Acción = $1A_{16}$
43			0000 0010	Sin extensión, Type = 2_{10},(Octet String)
44			0000 0010	Longitud del dato = 2_{10}
45			0001 1111	Inicio del contador = $1F_{16}$
46			0010 0100	Fin del contador = 24_{16}
	}			
47	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff**: valor de número PDU incrementado secuencialmente. Cuarto servicio concatenado (Opcional).
48	GET.request	SEQUENCE	0110	GET.request
	{			
	Indicador de Opción		0	Credencial de Acceso no presente
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeIdList presente
	Fill	BIT STRING(SIZE(1))	0	Poner en 0
49	EID	INTEGER(0..127,...)	**0**000 0010	Sin extensión, EID = 2 (elemento del emisor)
50	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
	AttributeId		**0**000 0001	Sin extensión, número de AttributeIDs = 1
	{			
51	Privado		**0**111 1101	AttributeId de atributo Privado = 125_{10} de configuración 4 de AIAs
	} } }			
52	FCS		xxxx xxxx	Secuencia verificadora de la trama
53			xxxx xxxx	
54	FLAG		0111 1110	"Flag" de término

C.2.6 Autenticación Fiscal y de Contrato en Transacción Nacional de Peaje Interoperable: GET_STAMPED.response, GET.response Opcional (Elemento del Emisor) y Ejemplo de GET.response Opcional (Elemento de Sistema) (ACn)

La respuesta GET_STAMPED.response es idéntica en las autenticaciones Fiscal y de Contrato, excepto por el valor de autenticación resultante en bytes 19 a 22. La primera respuesta opcional GET.response, bytes 23 al 35, sólo existe en el caso de Autenticación Fiscal, cuando se accede al Elemento reservado al emisor del transponder. La segunda respuesta opcional GET.response, bytes 36 al 51, también existe solamente en el caso de Autenticación Fiscal, cuando se accede al Elemento de Sistema del transponder.

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección Es “Up Link”
	MAC control field. R		0	No se solicita ventana privada de “Up Link”
	MAC control field. C/R		1	LPDU tipo respuesta
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	
	LLC control field. P/F		1	Bit Final = 1
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR		0000	Respuesta disponible
	LLC status field. CCCC		0000	Comando aceptado
9	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con primer servicio concatenado: GET_STAMPED.request.
10	GET_STAMPED.response	SEQUENCE	0001	ACTION.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	Parámetro de respuesta presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
11	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento de peaje Interop., relacionado con una ContextMark
12	ResponseParameter	CONTAINER	**0**001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{			
13	AttributeList	SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, cantidad de atributos: 1
	{			
	Attributes	SEQUENCE		
	{			
14	AttributeID		**0**001 1010	EquipmentStatus ID = 26_{10}
15	AttributeValue	CONTAINER	0011 1010	CHOICE: 58_{10} = EquipmentStatus
	{			
16	EquipmentStatus		ssss ssss	Valor de EquipmentStatus
17			ssss ssss	
	} } }			
18	Authenticator	OCTET STRING	**0**000 0100	Sin extensión. Longitud del “string”= 4 bytes
	{			
19	FiscalAuthenticator o ContractAutehnticator		xxxx xxxx	Autenticador calculado sobre EquipmentStatus, usando la clave ElementAuthenticationKey seleccionada por KeyRef, y el número aleatorio RndRSE.
20			xxxx xxxx	
21			xxxx xxxx	
22			xxxx xxxx	
	} } }			
23	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con 2º servicio concatenado: GET.request (opcional).
24	GET.response	SEQUENCE	0111	GET.response

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
25	EID	INTEGER(0..127,...)	0eee eeee	Sin extensión, EID del Elemento del emisor
26	attributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, 1 atributo en la lista.
	{			
27	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0110 0000	AttributeId = 96_{10} (Scratchpad)
28	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
29	{		0000 0110	Longitud de Scratchpad = 6_{10} bytes
30	Scratchpad		ssss ssss	Valor de Scratchpad
31			ssss ssss	
32			ssss ssss	
33			ssss ssss	
34			ssss ssss	
35			ssss ssss	
	} } } }			
36	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: GET.request (opcional).
37	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
38	EID	INTEGER(0..127,...)	0000 0000	Sin extensión, EID = 0 (Elemento de sistema)
39	attributeList	SEQUENCE (0..127,...) OF	0000 0010	Sin extensión, 2 atributos en la lista
	{			
40	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0000 0111	AttributeId de ActivityTimer de configuraciones 1, 2 y 3 de AIAs
41	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
42	{		0000 0100	Longitud de ActivityTimer = 4_{10} bytes
43	ActivityTimer		ssss ssss	Valor de ActivityTimer
44			ssss ssss	
45			ssss ssss	
46			ssss ssss	
47	}			
	AttributeId	INTEGER(0..127,...)	0001 0000	AttributeId de BatteryInsertionDate de configuraciones 1, 2 y 3 de AIAs
48	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
49	{		0000 0010	Longitud de BatteryInsertionDate = 2_{10} bytes
50	BatteryInsertionDate		dddd dddd	Valor de BatteryInsertionDate de configuraciones 1, 2 y 3 de AIAs
51			dddd dddd	
	}			

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
	} } }		
52	FCS	xxxx xxxx	Secuencia verificadora de la trama
53		xxxx xxxx	
54	FLAG	0111 1110	"Flag" de término

Cuando se leen los Atributos ActivityTimer y BatteryInsertionDate de la configuración 4 de AIAs, la respuesta tiene la codificación siguiente a partir del byte 36:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
36	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
37	ACTION.response	SEQUENCE	0001	PRIVATE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		x	0: ReturnStatus no presente, comando exitoso; 1: ReturnStatus presente si ocurrió un error
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
38	EID	INTEGER (0..127,...)	0000 0000	Sin extensión, EID de Elemento de sistema =0
39	Parameter	CONTAINER	0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
40	{		0000 0100	Sin extensión , largo de parámetro = 9 bytes
41	Action Type		0001 1010	Sin extensión , Tipo de Acción = $1A_{16}$
42	ActivityTimer		0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
43			0000 0110	Sin extensión , largo del dato = 6 bytes
44			cccc cccc	Valor de ActivityTimer
45			cccc cccc	
46			cccc cccc	
47			cccc cccc	
48			cccc cccc	
49			cccc cccc	
	} }			
50	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: GET.request (opcional).
51	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
52	EID	INTEGER(0..127,...)	0000 0000	Sin extensión, EID = 0 (Elemento de sistema)
53	attributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, 1 atributo en la lista
	{			
54	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0111 1101	AttributeId de Atributo Privado 125 de configuración 4 de AIAs

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
55	Attribute Value CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
56	**{**	0000 0110	Longitud de Atributo Privado 125 = 6_{10} bytes
57	Privado125.TransponderSerialNumber	nnnn nnnn	Valor de TransponderSerialNumber
58		nnnn nnnn	
59		nnnn nnnn	
60		nnnn nnnn	
61	Privado125.BatteryInsertionDate	aaaa aaas	Valor de BatteryInsertionDate
62		ssss srrr	
	}		
	} } }		
63	FCS	xxxx xxxx	Secuencia verificadora de la trama
64		xxxx xxxx	
65	FLAG	0111 1110	"Flag" de término

C.2.7 Autenticación Fiscal y de Contrato en Transacción Nacional de Peaje Interoperable: GET_STAMPED.response, GET.response Opcional (Elemento del Emisor) y Ejemplo de GET.response Opcional (Elemento de Sistema) (UI)

La respuesta GET_STAMPED.response es idéntica en las autenticaciones Fiscal y de Contrato, excepto por el valor de autenticación resultante en bytes 18 a 21. La primera respuesta opcional GET.response, bytes 22 al 34, sólo existe en el caso de Autenticación Fiscal, cuando se accede al Elemento reservado al emisor del transponder. La segunda respuesta opcional GET.response, bytes 35 al 50, también existe solamente en el caso de Autenticación Fiscal, cuando se accede al Elemento de sistema del transponder.

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección Es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. M	000	Comando UI
	LLC control field. P/F	0	No Poll
	LLC control field. M	01	Comando UI
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1
8	Fragmentation header	1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con primer servicio concatenado: GET_STAMPED.request
9	GET_STAMPED.response SEQUENCE	0001	ACTION.response

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	Parámetro de respuesta presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
10	EID	INTEGER (0..127,...)	0eee eeee	Sin extensión, EID de Elemento de peaje Interop., relacionado con una ContextMark
11	ResponseParameter	CONTAINER	0001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{			
12	AttributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, cantidad de atributos: 1
	{			
	Attributes	SEQUENCE		
	{			
13	AttributeID		0001 1010	EquipmentStatus ID = 26_{10}
14	AttributeValue	CONTAINER	0011 1010	CHOICE: 58_{10} = EquipmentStatus
	{			
15	EquipmentStatus		ssss ssss	Valor de EquipmentStatus
16			ssss ssss	
	} } }			
17	Authenticator	OCTET STRING	0000 0100	Sin extensión. Longitud del "string"= 4 bytes
	{			
18	FiscalAuthenticator o ContractAutehnticator		xxxx xxxx	Autenticador calculado sobre EquipmentStatus, usando la clave ElementAuthenticationKey seleccionada por KeyRef, y el número aleatorio RndRSE.
19			xxxx xxxx	
20			xxxx xxxx	
21			xxxx xxxx	
	} } }			
22	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con 2º servicio concatenado: GET.request (opcional).
23	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
24	EID	INTEGER(0..127,...)	0eee eeee	Sin extensión, EID del Elemento del emisor
25	attributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, 1 atributo en la lista.
	{			
26	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0110 0000	AttributeId = 96_{10} (Scratchpad)
27	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
28	{		0000 0110	Longitud de Scratchpad = 6_{10} bytes
29	Scratchpad		ssss ssss	Valor de Scratchpad
30			ssss ssss	
31			ssss ssss	
32			ssss ssss	
33			ssss ssss	
34			ssss ssss	
	} } } }			

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
35	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: GET.request (opcional).
36	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
37	EID	INTEGER(0..127,...)	0000 0000	Sin extensión, EID = 0 (Elemento de sistema)
38	attributeList	SEQUENCE (0..127,...) OF	0000 0010	Sin extensión, 2 atributos en la lista.
	{			
39	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	0aaa aaaa	AttributeId de ActivityTimer de configuraciones 1, 2 y 3 de AIAs
40	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
41	{		0000 0100	Longitud de ActivityTimer = 4_{10} bytes
42	ActivityTimer		ssss ssss	Valor de ActivityTimer
43			ssss ssss	
44			ssss ssss	
45			ssss ssss	
46	}			
	AttributeId	INTEGER(0..127,...)	0001 0000	AttributeId de BatteryInsertionDate de configuraciones 1, 2 y 3 de AIAs
47	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
48	{		0000 0010	Longitud de BatteryInsertionDate = 2_{10} bytes
49	BatteryInsertionDate		dddd dddd	Valor de BatteryInsertionDate
50			dddd dddd	
	}			
	} } }			
51	FCS		xxxx xxxx	Secuencia verificadora de la trama
52			xxxx xxxx	
53	FLAG		0111 1110	"Flag" de término

Cuando se leen los Atributos ActivityTimer y BatteryInsertionDate de la configuración 4 de AIAs, la respuesta tiene la codificación siguiente a partir del byte 35:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
35	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
36	ACTION.response	SEQUENCE	0001	PRIVATE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		x	0: ReturnStatus no presente, comando exitoso; 1: ReturnStatus presente si ocurrió un error

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
37	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID de Elemento de sistema =0
38	Parameter	CONTAINER	**0**000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
39	**{**		**0**000 0100	Sin extensión , largo de parámetro = 9 bytes
40	Action Type		**0**001 1010	Sin extensión , Tipo de Acción = $1A_{16}$
41	ActivityTimer		0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
42			**0**000 0110	Sin extensión , largo del dato = 6 bytes
43			cccc cccc	Valor de ActivityTimer
44			cccc cccc	
45			cccc cccc	
46			cccc cccc	
47			cccc cccc	
48			cccc cccc	
	} }			
49	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: GET.request (opcional).
50	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
51	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de sistema)
52	attributeList	SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, 1 atributo en la lista
	{			
53	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	**0**111 1101	AttributeId de Atributo Privado 125 de configuración 4 de AIAs
54	Attribute Value	CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10} (Octet string)
55	**{**		0000 0110	Longitud de Atributo Privado 125 = 6_{10} bytes
56	Privado125.TransponderSerialNr.		nnnn nnnn	Valor de TransponderSerialNumber
57			nnnn nnnn	
58			nnnn nnnn	
59			nnnn nnnn	
60	Privado125.BatteryInsertionDate		aaaa aaas	Valor de BatteryInsertionDate
61			ssss srrr	
	}			
	} } }			
62	FCS		xxxx xxxx	Secuencia verificadora de la trama
63			xxxx xxxx	
64	FLAG		0111 1110	"Flag" de término

C.2.8 RECIBO EN TRANSACCIÓN NACIONAL DE PEAJE INTEROPERABLE: SERVICIOS CONCATENADOS: SET.REQUEST, SET.REQUEST OPCIONAL, SET. REQUEST OPCIONAL PARA BAJAR EL TAMPER BIT Y SET_MMI.REQUEST (ACn)

Los bytes 36 al 42 son optativos, sólo se incluyen cuando se utiliza el Atributo Receipt Authenticator. Lo mismo vale para los bytes 47 a 62, que están presentes si se utiliza el Atributo Spare. Asimismo, el servicio SET.request en los bytes 63 a 80 es opcional. Está presente sólo cuando se accede a la información contenida en el Elemento reservado al emisor del transponder. El servicio SET.request en los bytes 81 al 94, también opcional, muestra la forma en que se aplica el reset al bit de Tamper.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx0	
4			xxxx xxx0	
5			xxxx xxx1	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	Comando ACn 1 = Poll, 0 = no Poll
	LLC control field. P/F		1	
	LLC control field. M		01	
	LLC control field. reserved bits		11	Bits no usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Primer servicio concatenado.
9	SET.request	SEQUENCE	0100	SET.request
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		0	IID no presente
	Fill	BIT STRING(SIZE(1))	0	Llenar con 0
	Mode	BOOLEAN	1	Modo confirmado = 1, respuesta esperada
10	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento de peaje Interoperable
11	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{			
12	AC_CR		aaaa aaaa	Credencial de acceso calculada por RSE usando RndOBE y la clave de acceso ala elemento EAcKey del elemento de peaje Interoperable
13			aaaa aaaa	
14			aaaa aaaa	
15			aaaa aaaa	
	}			
16	AttributeList	SEQUENCE ((0..127,...) OF		
	{			
	Attributes	SEQUENCE	**0**000 0xxx	Sin extens., número de atributos en lista = 3_{10},

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			4_{10} o 5_{10}, dependiendo si se escribe o no en los atributos ReceiptAuthenticator y/o Spare
17	AttributeId	INTEGER(0..127,...)	0000 0101	AttributeId = 5_{10} (ReceiptServicePart)
18	Attribute Value	CONTAINER	0010 0101	CONTAINER CHOICE = 37_{10}
	{			
19	ReceiptServicePart		xxxx xxxx	Valor de ReceiptServicePart.SessionTime
20			xxxx xxxx	
21			xxxx xxxx	
22			xxxx xxxx	
23			xxxx xxxx	Valor de ReceiptServicePart.Provider
24			xxxx xxxx	
25			xxxx xxxx	
26			xxxx xxxx	Valor de ReceiptServicePart.StationLocation
27			xxxx xxxx	
28			xxxx	
			xxxx	Valor de ReceiptServicePart.SessionLocation
29			xxxx	
			xxxx	Valor de ReceiptServicePart.TypeOfSession
30			xxxx xxxx	ReceiptServicePart.SessionResultOperational
31			xxxx xxxx	ReceiptServicePart.SessionResultFinancial
	}			
32	AttributeId	INTEGER(0..127,...)	0000 0110	AttributeId = 6_{10} (SessionClass)
33	Attribute Value	CONTAINER	0010 0110	CONTAINER CHOICE = 38_{10}
	{			
34	SessionClass		xxxx xxxx	Valor de SessionClass.SessionTariffClass
35			xxxx xxxx	Valor de SessionClass.SessionClaimedClass
	}			
36	AttributeId	INTEGER(0..127,...)	0000 1101	AttributeId = 13_{10} (ReceiptAuthenticator)
37	Attribute Value	CONTAINER	0010 1101	CONTAINER CHOICE = 45_{10}
	{			
38	ReceiptAuthenticator		0000 0100	Longitud de ReceiptAuthenticator = 4_{10}
39			aaaa aaaa	Valor de ReceiptAuthenticator
40			aaaa aaaa	
41			aaaa aaaa	
42			aaaa aaaa	
	}			
43	AttributeId	INTEGER(0..127,...)	0001 1010	AttributeId = 26_{10} (EquipmentStatus)
44	Attribute Value	CONTAINER	0011 1010	CONTAINER CHOICE = 58_{10}
	{			
45	EquipmentStatus		xxxx xxxx	Valor de EquipmentStatus
46			xxxx xxxx	
	}			
47	AttributeId	INTEGER(0..127,...)	0110 0010	AttributeId = 98_{10} (Spare)
48	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10}
49	{		0000 1101	Longitud de Spare = 13_{10} bytes
50	Spare		xxxx xxxx	Valor de Spare
51			xxxx xxxx	
52			xxxx xxxx	
53			xxxx xxxx	
54			xxxx xxxx	

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
55			xxxx xxxx	
56			xxxx xxxx	
57			xxxx xxxx	
58			xxxx xxxx	
59			xxxx xxxx	
60			xxxx xxxx	
61			xxxx xxxx	
62			xxxx xxxx	
	}			
	} } }			
63	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Segundo servicio concatenado (opcional).
64	SET.request	SEQUENCE	0100	SET.request
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		0	IID no presente
	Fill	BIT STRING(SIZE(1))	0	Llenar con 0
	Mode	BOOLEAN	1	Modo confirmado = 1, respuesta esperada
65	EID	INTEGER(0..127,...)	0eee eeee	Sin extensión, EID del Elemento del emisor
66	AccessCredential	OCTET STRING	0000 0100	Sin extensión, longitud del “string”= 4 bytes
	{			
67	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando el ElementAccessKey EAcKey del Elemento del emisor del transponder y: RndOBE, en configuraciones 1, 3 y 4 de AIAs Nonce, en configuración 2 de AIAs
68			aaaa aaaa	
69			aaaa aaaa	
70			aaaa aaaa	
	}			
71	AttributeList	SEQUENCE ((0..127,...) OF		
	{			
	Attributes	SEQUENCE	0000 0001	Sin extensión, número de atributos en lista = 1
	{			
72	AttributeId	INTEGER(0..127,...)	0110 0000	AttributeId = 96_{10} (Scratchpad)
73	Attribute Value	CONTAINER	0000 0010	CONTAINER CHOICE = 2_{10}
74	{		0000 0110	Longitud de Scratchpad = 6_{10} bytes
75	Scratchpad		xxxx xxxx	Valor de Scratchpad
76			xxxx xxxx	
77			xxxx xxxx	
78			xxxx xxxx	
79			xxxx xxxx	
80			xxxx xxxx	
	} } } }			
81	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Tercer servicio concatenado (opcional).
82	SET.request	SEQUENCE	0100	SET.request
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		0	IID no presente
	Fill	BIT STRING(SIZE(1))	0	Llenar con 0
	Mode	BOOLEAN	1	Modo confirmado = 1, respuesta esperada

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
83	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID del Elemento de sistema
84	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string"= 4 bytes
	{			
85	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando el ElementAccessKey EAcKey del Elemento de sistema y: RndOBE, en configuraciones 1 y 3 de AIAs Nonce, en configuración 2 de AIAs
86			aaaa aaaa	
87			aaaa aaaa	
88			aaaa aaaa	
	}			
89	AttributeList	SEQUENCE ((0..127,...) OF		
	{			
	Attributes	SEQUENCE	**0**000 0001	Sin extensión, número de atributos en lista = 1
	{			
90	AttributeId	INTEGER(0..127,...)	**0**000 1010	AttributeId = 10_{10} (obeStatus)
91	Attribute Value	CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10}
92	**{**		0000 0110	Longitud de obeStatus = 2_{10} bytes
93	obeStatus		xxxx 0xxx	Valor de obeStatus. Se aplica reset al bit de Tamper.
94			xxxx xxxx	
	} } } }			
95	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Cuarto servicio concatenado.
96	SET_MMI.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		0	Credencial de acceso no presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
97	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
98	ActionType	INTEGER(0..127,...)	**0**000 1010	Sin extensión, SET_MMI.request = 10_{10}
99	ActionParameter	CONTAINER	**0**000 0000	Sin extensión, Type 0 = INTEGER
100	SetMMI	INTEGER	0000 00mm	0: OK, 1: no OK, 2: contactar al operador
	}			
101	FCS		xxxx xxxx	Secuencia verificadora de la trama
102			xxxx xxxx	
103	FLAG		0111 1110	"Flag" de término

Para aplicar reset al bit de Tamper de la configuración 4 de AIAs, se usa la siguiente codificación a partir del byte 81:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
81	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Tercer servicio concatenado (alternativo para configuración 4 de AIAs).
82	PRIVATE.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		1	Credencial de acceso presente

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
83	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID del Elemento de sistema
84	ActionType	INTEGER(0..127,...)	**0**111 0111	Sin extensión, PRIVATE.request = 119_{10}
85	AccessCredential	OCTET STRING	**0**000 0101	Sin extensión, largo de AC_CR = 5_{10} bytes
	{			
86	AC_CR		aaaa aaaa	Estos 4 bytes de la Credencial de Acceso son determinados por el RSE usando el procedimiento descrito en 6.2, excepto que en lugar de EAcK se usa la clave TampK.
87			aaaa aaaa	
88			aaaa aaaa	
89			aaaa aaaa	
90			0000 1000	08_{16}
	}			
91	Action Parameter	CONTAINER	0000 0010	Sin extensión, Type = 2_{10},(Octet String)
92			0000 0101	Longitud del parámetro = 5_{10}
93			0010 0001	Tipo de Acción = 21_{16}
94			0000 0010	Sin extensión, Type = 2_{10},(Octet String)
95			0000 0010	Longitud del dato = 2_{10}
96			0000 0000	Inicio del dato a leer = 00_{16}
97			0000 0111	Fin del dato a leer = 07_{16}
	}			
98	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente. Cuarto servicio concatenado.
99	SET_MMI.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		0	Credencial de acceso no presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
100	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
101	ActionType	INTEGER(0..127,...)	**0**000 1010	Sin extensión, SET_MMI.request = 10_{10}
102	ActionParameter	CONTAINER	**0**000 0000	Sin extensión, Type 0 = INTEGER
103	SetMMI	INTEGER	0000 00mm	0: OK, 1: no OK, 2: contactar al operador
	}			
104	FCS		xxxx xxxx	Secuencia verificadora de la trama
105			xxxx xxxx	
106	FLAG		0111 1110	"Flag" de término

C.2.9 Recibo en Transacción Nacional de Peaje Interoperable: Servicios Concatenados: SET.response, SET.response Opcional, SET.response Opcional de Bajada del Bit de Tamper y SET_MMI.response (ACn)

Los bytes 12 a 14 sólo están presentes cuando se accede a la información contenida en el Elemento reservado al emisor del transponder. Los bytes 15 a 17 sólo están presentes cuando se ha escrito en elemento de sistema.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		1	LPDU tipo respuesta
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	
	LLC control field. P/F		1	Bit final = 1
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR		0000	Respuesta disponible
	LLC status field. CCCC		0000	Comando aceptado
9	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con primer servicio concatenado: SET.request.
10	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
11	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento de peaje Interoperable
	}			
12	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con segundo servicio concatenado: SET.request. (opcional).
13	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
14	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento del emisor
	}			
15	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: SET.request. (opcional).
16	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
17	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID del Elemento de sistema
	}			
18	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: ACTION.request.
19	ACTION.response	SEQUENCE	0001	SET_MMI.response

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ResponseParameter no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
20	EID	INTEGER (0..127,...)	0000 0000	Sin extensión, EID de Elemento de sistema =0
	}			
21	FCS		xxxx xxxx	Secuencia verificadora de la trama
22			xxxx xxxx	
23	FLAG		0111 1110	"Flag" de término

Cuando se aplica el reset al bit de Tamper de la configuración 4 de AIAs, la respuesta tiene la codificación siguiente a partir del byte 15:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
15	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
16	ACTION.response	SEQUENCE	0001	PRIVATE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		x	0: ReturnStatus no presente, comando exitoso; 1: ReturnStatus presente si ocurrió un error
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
17	EID	INTEGER (0..127,...)	0000 0000	Sin extensión, EID de Elemento de sistema =0
18	Parameter	CONTAINER	0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
19	{		0000 1011	Sin extensión , largo de parámetro = B_{16} bytes
20	Action Type		0010 0001	Sin extensión , Tipo de Acción = 21_{16}
21			0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
22			0000 1000	Sin extensión , largo del dato = 8 bytes
23			0000 0000	Dato artificial, no se usa
24			0000 0000	
25			0000 0000	
26			0000 0000	
27			0000 0000	
28			0000 0000	
29			0000 0000	
30			0000 0000	
	} }			
31	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: ACTION.request.
32	ACTION.response	SEQUENCE	0001	SET_MMI.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ResponseParameter no presente

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
33	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID de Elemento de sistema =0
	}			
34	FCS		xxxx xxxx	Secuencia verificadora de la trama
35			xxxx xxxx	
36	FLAG		0111 1110	"Flag" de término

C.2.10 RECIBO EN TRANSACCIÓN NACIONAL DE PEAJE INTEROPERABLE: SERVICIOS CONCATENADOS: SET.RESPONSE, SET.RESPONSE OPCIONAL, SET.RESPONSE OPCIONAL DE BAJADA DEL BIT DE TAMPER Y SET_MMI.RESPONSE (UI)

Los bytes 11 a 13 sólo están presentes cuando se accede a la información contenida en el Elemento reservado al emisor del transponder. Los bytes 14 a 16 sólo están presentes cuando se ha escrito en el elemento de sistema.

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	No Poll
	LLC control field. M		01	Comando UI
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con primer servicio concatenado: SET.request
9	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
10	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento de peaje Interoperable
	}			
11	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con 2º servicio concatenado: SET.request (opcional).
12	SET.response	SEQUENCE	0101	SET.response
	{			

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
13	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento del emisor
	}			
14	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: SET.request (opcional).
15	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
16	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID del Elemento de sistema
	}			
17	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: ACTION.request.
18	ACTION.response	SEQUENCE	0001	SET_MMI.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ResponseParameter no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
19	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID de Elemento de sistema =0
	}			
20	FCS		xxxx xxxx	Secuencia verificadora de la trama
21			xxxx xxxx	
22	FLAG		0111 1110	"Flag" de término

Cuando se aplica el reset al bit de Tamper de la configuración 4 de AIAs, la respuesta tiene la codificación siguiente a partir del byte 14:

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
14	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con tercer servicio concatenado: ACTION.request.
15	ACTION.response	SEQUENCE	0001	PRIVATE.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	ResponseParameter presente
	Indicador de Opción		x	0: ReturnStatus no presente, comando exitoso; 1: ReturnStatus presente si ocurrió un error
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
16	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID de Elemento de sistema =0
17	Parameter	CONTAINER	**0**000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
18	**{**		**0**000 1011	Sin extensión , largo de parámetro = B_{16} bytes
19	Action Type		**0**010 0001	Sin extensión , Tipo de Acción = 21_{16}

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
20			0000 0010	CONTAINER TYPE = 2_{10} (OCTET STRING)
21			**0**000 1000	Sin extensión , largo del dato = 8 bytes
22			0000 0000	Dato falso, no se usa
23			0000 0000	
24			0000 0000	
25			0000 0000	
26			0000 0000	
27			0000 0000	
28			0000 0000	
29			0000 0000	
	} }			
30	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con cuarto servicio concatenado: ACTION.request.
31	ACTION.response	SEQUENCE	0001	SET_MMI.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ResponseParameter no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
32	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID de Elemento de sistema =0
	}			
33	FCS		xxxx xxxx	Secuencia verificadora de la trama
34			xxxx xxxx	
35	FLAG		0111 1110	"Flag" de término

C.3 PRESENTACIÓN Y RECIBO EN TRANSACCIÓN DE GESTIÓN DE ESTACIONAMIENTOS

C.3.1 PRESENTACIÓN : GET_STAMPED.REQUEST (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx0	
4			xxxx xxx0	
5			xxxx xxx1	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	Comando ACn
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente.
9			0000	ACTION.request
	GET_STAMPED.request	SEQUENCE		
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
10	EID	INTEGER(0..127,...)	**0**eee eeee	EID del Elemento de G. de Estacionamientos, relacionado con una ContextMark.
11	ActionType	INTEGER(0..127,...)	**0**000 0000	Sin extensión, GET_STAMPED.request = 0
12	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{			
13	AC_CR		aaaa aaaa	Credencial de acceso calculada por RSE usando RndOBE y la clave de acceso EAcKey del Elemento de Gestión de Estacionamientos.
14			aaaa aaaa	
15			aaaa aaaa	
16			aaaa aaaa	
	}			
17	ActionParameter	CONTAINER	**0**001 0001	Sin extensión, CHOICE 17_{10} = GetStampedRq
	{			
	AttributeIdList	AttrIdList		
	{			
18		SEQUENCE (0..127,...)	**0**000 0001	Sin extensión, cantidad de AttributeIDs = 1
	{			
19	ContractSerialNumber		0000 0001	ContractSerialNumber ID = 1_{10}

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
	} }		
20	nonce OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{		
21	RndRSE	rrrr rrrr	Número aleatorio generado por el RSE, necesario para calcular el valor de ContractAuthenticator
22		rrrr rrrr	
23		rrrr rrrr	
24		rrrr rrrr	
	}		
25	KeyRef	**0**110 1111	Referencia a clave de autenticación usada en el cálculo del ContractAuthenticator: EAuK = 111_{10}
	} }		
26	FCS	xxxx xxxx	Secuencia verificadora de la trama
27		xxxx xxxx	
28	FLAG	0111 1110	"Flag" de término

C.3.2 PRESENTACIÓN: GET_STAMPED.RESPONSE (ACn)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	1	LPDU tipo respuesta
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. n	N	Bit n de comando ACn
	LLC control field. M	11	
	LLC control field. P/F	1	Bit final = 1
	LLC control field. M	01	
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR	0000	Respuesta disponible
	LLC status field. CCCC	0000	Comando aceptado
9	Fragmentation header	1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio GET_STAMPED.request.
10	GET_STAMPED.response SEQUENCE	0001	ACTION.response
	{		
	Indicador de Opción	0	IID no presente
	Indicador de Opción	1	Parámetro de respuesta presente
	Indicador de Opción	0	ReturnStatus no presente
	Fill BIT STRING (SIZE(1))	0	Llenar con 0
11	EID INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento de Gestión de Estacionamientos, relacionado con una ContextMark.

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
12	ResponseParameter CONTAINER	**0**001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{		
13	AttributeList SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, cantidad de atributos = 1
	{		
	Attributes SEQUENCE		
	{		
14	AttributeID	**0**000 0001	ContractSerialNumber ID = 1_{10}
15	AttributeValue CONTAINER	0010 0001	CHOICE: 33_{10} = ContractSerialNumbr
	{		
16	ContractSerialNumber	aaaa aaaa	Valor de ContractSerialNumber
17		aaaa aaaa	
18		aaaa aaaa	
19		aaaa aaaa	
	} } }		
20	Authenticator OCTET STRING	**0**000 0100	Sin extensión, longitud del "string"= 4 bytes
	{		
21	ContractAuthenticator	xxxx xxxx	Autenticador calculado sobre ContractSerialNumber, usando la clave ElementAuthenticationKey seleccionada por KeyRef, y el número aleatorio RndRSE.
22		xxxx xxxx	
23		xxxx xxxx	
24		xxxx xxxx	
	} } }		
25	FCS	xxxx xxxx	Secuencia verificadora de la trama
26		xxxx xxxx	
27	FLAG	0111 1110	"Flag" de término

C.3.3 PRESENTACIÓN: GET_STAMPED.RESPONSE (UI)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. M	000	Comando UI
	LLC control field. P/F	0	No Poll
	LLC control field. M	01	Comando UI
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	Fragmentation header	1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio GET_STAMPED.request.
9	GET_STAMPED.response SEQUENCE	0001	ACTION.response
	{		

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	Parámetro de respuesta presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
10	EID	INTEGER (0..127,...)	0eee eeee	Sin extensión, EID del Elemento de Gestión de Estacionamientos, relacionado con una ContextMark.
11	ResponseParameter	CONTAINER	0001 0010	Sin extensión. CHOICE 18_{10} = GetStampedRs
	{			
12	AttributeList	SEQUENCE (0..127,...) OF	0000 0001	Sin extensión, cantidad de atributos = 1
	{			
	Attributes	SEQUENCE		
	{			
13	AttributeID		0000 0001	ContractSerialNumber ID = 1_{10}
14	AttributeValue	CONTAINER	0010 0001	CHOICE: 33_{10} = ContractSerialNumbr
	{			
15	ContractSerialNumber		aaaa aaaa	Valor de ContractSerialNumber
16			aaaa aaaa	
17			aaaa aaaa	
18			aaaa aaaa	
	} } }			
19	Authenticator	OCTET STRING	0000 0100	Sin extensión, longitud del “string”= 4 bytes
	{			
20	ContractAuthenticator		xxxx xxxx	Autenticador calculado sobre ContractSerialNumber, usando la clave ElementAuthenticationKey seleccionada por KeyRef, y el número aleatorio RndRSE.
21			xxxx xxxx	
22			xxxx xxxx	
23			xxxx xxxx	
	} } }			
24	FCS		xxxx xxxx	Secuencia verificadora de la trama
25			xxxx xxxx	
26	FLAG		0111 1110	“Flag” de término

C.3.4 RECIBO : SET_MMI.REQUEST (ACn)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	“Flag” inicial
2	Private LID	xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx0	
4		xxxx xxx0	
5		xxxx xxx1	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	0	Dirección es “Down Link”
	MAC control field. A	1	RSE asigna ventana privada en el “Up Link”
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. S	S	Bit de secuencia
	MAC control field. reserved bits	000	Bits reservados
7	LLC control field. n	N	Bit n de comando ACn

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	LLC control field. M		11	Comando ACn
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
26	Fragmentation header		1**fff f**001	Sin fragmentación,. **ffff:** valor de número PDU incrementado secuencialmente.
27	SET_MMI.request	SEQUENCE	0000	ACTION.request
	{			
	Indicador de Opción		0	Credencial de acceso no presente
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
28	EID	INTEGER(0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
29	ActionType	INTEGER(0..127,...)	**0**000 1010	Sin extensión, SET_MMI.request = 10_{10}
30	ActionParameter	CONTAINER	**0**000 0000	Sin extensión, tipo 0 = INTEGER
31	SetMMI	INTEGER	0000 00mm	0: OK, 1: no OK, 2: contactar al operador
	}			
32	FCS		xxxx xxxx	Secuencia verificadora de la trama
33			xxxx xxxx	
34	FLAG		0111 1110	"Flag" de término

C.3.5 RECIBO: SET_MMI.RESPONSE (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE
3			xxxx xxx**0**	específico
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		1	LPDU tipo respuesta
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	
	LLC control field. P/F		1	Bit final = 1
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR		0000	Respuesta disponible
	LLC status field. CCCC		0000	Comando aceptado
9	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio ACTION.request.
10	ACTION.response	SEQUENCE	0001	SET_MMI.response
	{			
	Indicador de Opción		0	IID no presente

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
	Indicador de Opción		0	Parámetro de la respuesta no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
11	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
	}			
12	FCS		xxxx xxxx	Secuencia verificadora de la trama
13			xxxx xxxx	
14	FLAG		0111 1110	"Flag" de término

C.3.6 RECIBO:SET_MMI.RESPONSE (UI)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	No Poll
	LLC control field. M		01	Comando UI
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio ACTION.request.
9	ACTION.response	SEQUENCE	0001	SET_MMI.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	Parámetro de la respuesta no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
10	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
	}			
11	FCS		xxxx xxxx	Secuencia verificadora de la trama
12			xxxx xxxx	
13	FLAG		0111 1110	"Flag" de término

C.4 PRESENTACIÓN Y ACTUALIZACIÓN EN TRANSACCIÓN DE SONDA DE TRÁFICO

C.4.1 PRESENTACIÓN: GET.REQUEST (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx0	
4			xxxx xxx0	
5			xxxx xxx1	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	Comando ACn
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1fff f001	Sin fragmentación. **ffff:** valor de número PDU incrementado secuencialmente.
9	GET.request	SEQUENCE	0110	GET.request
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeIdList presente
	Fill	BIT STRING(SIZE(1))	0	Llenar con 0
10	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento para Sonda de Tráfico
11	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{			
12	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando RndOBE y la clave de acceso EAcKey del Elemento para Sonda de Tráfico.
13			aaaa aaaa	
14			aaaa aaaa	
15			aaaa aaaa	
	}			
16	AttributeIdList	SEQUENCE (0..127,...) OF		
	{	INTEGER (0..127,...)		
	AttributeId		**0**000 0001	Sin extensión, cantidad de AttributeIDs = 1
	{			
17	TemporaryID		0110 0001	attributeID = 97_{10} (TemporaryID)
	} } }			
18	FCS		xxxx xxxx	Secuencia verificadora de la trama
19			xxxx xxxx	
20	FLAG		0111 1110	"Flag" de término

C.4.2 PRESENTACIÓN: GET.RESPONSE (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		1	LPDU tipo respuesta
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	
	LLC control field. P/F		1	Bit final = 1
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR		0000	Respuesta disponible
	LLC status field. CCCC		0000	Comando aceptado
9	Fragmentation header		1**fff** f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio GET.request.
10	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
11	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento para Sonda de Tráfico
12	AttributeList	SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, 1 atributo en la lista.
	{			
13	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	**0**110 0001	AttributeId = 97_{10} (TemporaryID)
14	Attribute Value	CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10}
15	**{**		0000 0011	Longitud de TemporaryID = 3 bytes
16	TemporaryID		aaaa aaaa	Valor de TemporaryID
17			aaaa aaaa	
18			aaaa aaaa	
	} } } }			
19	FCS		xxxx xxxx	Secuencia verificadora de la trama
20			xxxx xxxx	
21	FLAG		0111 1110	"Flag" de término

C.4.3 PRESENTACIÓN: GET.RESPONSE (UI)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es "Up Link"
	MAC control field. R		0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	No Poll
	LLC control field. M		01	Comando UI
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio GET.request.
9	GET.response	SEQUENCE	0111	GET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		1	AttributeList presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(1))	0	Llenar con 0
10	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento para Sonda de Tráfico
11	AttributeList	SEQUENCE (0..127,...) OF	**0**000 0001	Sin extensión, 1 atributo en la lista.
	{			
12	Attributes	SEQUENCE		
	{			
	AttributeId	INTEGER(0..127,...)	**0**110 0001	AttributeId = 97_{10} (TemporaryID)
13	Attribute Value	CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10}
14	**{**		0000 0011	Longitud de TemporaryID = 3 bytes
15	TemporaryID		aaaa aaaa	Valor de TemporaryID
16			aaaa aaaa	
17			aaaa aaaa	
	} } } }			
18	FCS		xxxx xxxx	Secuencia verificadora de la trama
19			xxxx xxxx	
20	FLAG		0111 1110	"Flag" de término

C.4.4 ACTUALIZACIÓN: SET.REQUEST (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx0	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx0	
4			xxxx xxx0	
5			xxxx xxx1	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	Comando ACn
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1fff f001	Sin fragmentación, **ffff:** valor de número PDU incrementado secuencialmente.
9	SET.request	SEQUENCE	0100	SET.request
	{			
	Indicador de Opción		1	Credencial de acceso presente
	Indicador de Opción		0	IID no presente
	Fill	BIT STRING(SIZE(1))	0	Llenar con 0
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
10	EID	INTEGER(0..127,...)	**0**eee eeee	Sin extensión, EID de Elemento para Sonda de Tráfico.
11	AccessCredential	OCTET STRING	**0**000 0100	Sin extensión, longitud del "string" = 4_{10} bytes
	{			
12	AC_CR		aaaa aaaa	Credencial de acceso calculada por el RSE usando RndOBE y la clave de acceso EAcKey del Elemento para Sonda de Tráfico.
13			aaaa aaaa	
14			aaaa aaaa	
15			aaaa aaaa	
	}			
16	AttributeList	SEQUENCE ((0..127,...) OF		
	{			
	Attributes	SEQUENCE	**0**000 0001	Sin extensión, 1 atributo en la lista
	{			
17	AttributeId	INTEGER(0..127,...)	**0**110 0001	AttributeId = 97_{10} (TemporaryID)
18	Attribute Value	CONTAINER	**0**000 0010	CONTAINER CHOICE = 2_{10}
19	{		**0**000 0011	Longitud de TemporaryID = 3 bytes
20	TemporaryID		xxxx xxxx	Valor de TemporaryID
21			xxxx xxxx	
22			xxxx xxxx	
	} } } }			
19	FCS		xxxx xxxx	Secuencia verificadora de la trama
20			xxxx xxxx	

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
21	FLAG	0111 1110	"Flag" de término

C.4.5 ACTUALIZACIÓN: SET.RESPONSE (ACn)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es "Up Link"
	MAC control field. R	0	No se solicita ventana privada de "Up Link"
	MAC control field. C/R	1	LPDU tipo respuesta
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. n	N	Bit n de comando ACn
	LLC control field. M	11	
	LLC control field. P/F	1	Bit final = 1
	LLC control field. M	01	
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR	0000	Respuesta disponible
	LLC status field. CCCC	0000	Comando aceptado
9	Fragmentation header	1**fff** f001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio SET.request.
10	SET.response SEQUENCE	0101	SET.response
	{		
	Indicador de Opción	0	IID no presente
	Indicador de Opción	0	ReturnStatus no presente
	Fill BIT STRING (SIZE(2))	00	Llenar con 0
11	EID INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento para Sonda de Tráfico
	}		
12	FCS	xxxx xxxx	Secuencia verificadora de la trama
13		xxxx xxxx	
14	FLAG	0111 1110	"Flag" de término

C.4.6 ACTUALIZACIÓN: SET.RESPONSE (UI)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		1	Dirección es “Up Link”
	MAC control field. R		0	No se solicita ventana privada de “Up Link”
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. reserved bits		0000	Bits reservados
7	LLC control field. M		000	Comando UI
	LLC control field. P/F		0	No Poll
	LLC control field. M		01	Comando UI
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación, **ffff:** mismo valor de número PDU recibido con servicio GET.request.
9	SET.response	SEQUENCE	0101	SET.response
	{			
	Indicador de Opción		0	IID no presente
	Indicador de Opción		0	ReturnStatus no presente
	Fill	BIT STRING (SIZE(2))	00	Llenar con 0
10	EID	INTEGER (0..127,...)	**0**eee eeee	Sin extensión, EID del Elemento para Sonda de Tráfico
	}			
11	FCS		xxxx xxxx	Secuencia verificadora de la trama
12			xxxx xxxx	
13	FLAG		0111 1110	“Flag” de término

C.5 TÉRMINO DE LA TRANSACCIÓN

C.5.1 TRACKING: ECHO.REQUEST (ACn)

Byte #	Atributo / Campo		Bits b_7 b_0	Descripción
1	FLAG		0111 1110	"Flag" inicial
2	Private LID		xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3			xxxx xxx**0**	
4			xxxx xxx**0**	
5			xxxx xxx**1**	
6	MAC control field. L		1	La trama contiene un LPDU
	MAC control field. D		0	Dirección es "Down Link"
	MAC control field. A		1	RSE asigna ventana privada en el "Up Link"
	MAC control field. C/R		0	LPDU tipo comando
	MAC control field. S		S	Bit de secuencia
	MAC control field. reserved bits		000	Bits reservados
7	LLC control field. n		N	Bit n de comando ACn
	LLC control field. M		11	
	LLC control field. P/F		1	1 = Poll, 0 = no Poll
	LLC control field. M		01	
	LLC control field. reserved bits		11	No usados. Mantenerlos en 1.
8	Fragmentation header		1**fff f**001	Sin fragmentación. **ffff:** valor de número PDU incrementado secuencialmente.
9			0000	ACTION.request
	ECHO.request	SEQUENCE		
	{			
	Indicador de Opción		0	Sin credencial de acceso
	Indicador de Opción		1	ActionParameter presente
	Indicador de Opción		0	IID no presente
	Mode	BOOLEAN	1	Modo confirmado, respuesta esperada
10	EID	INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID = 0
11	ActionType	INTEGER (0..127,...)	**0**000 1111	Sin extensión, ECHO.request = 15
12	ActionParameter	CONTAINER	**0**000 0010	Sin extensión, CHOICE 2_{10} = OCTET STRING
13			**0**000 0000	Sin extensión. Longitud del "string" = 0 bytes
	}			
14	FCS		xxxx xxxx	Secuencia verificadora de la trama
15			xxxx xxxx	
16	FLAG		0111 1110	"Flag" de término

C.5.2 TRACKING: ECHO.RESPONSE (ACn)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	"Flag" inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		Xxxx xxx**0**	

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
4		Xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	1	Dirección es “Up Link”
	MAC control field. R	0	No se solicita ventana privada de “Up Link”
	MAC control field. C/R	1	LPDU tipo respuesta
	MAC control field. reserved bits	0000	Bits reservados
7	LLC control field. n	N	Bit n de comando ACn
	LLC control field. M	11	
	LLC control field. P/F	1	Bit final = 1
	LLC control field. M	01	
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	LLC status field. RRRR	0000	Respuesta disponible
	LLC status field. CCCC	0000	Comando aceptado
9	Fragmentation header	1**fff f**001	Sin fragmentación. **ffff:** mismo valor de número PDU recibido con ECHO.request.
10		0001	ACTION.response
	ECHO.response SEQUENCE		
	{		
	Indicador de Opción	0	IID no presente
	Indicador de Opción	1	Parámetro de respuesta presente
	Indicador de Opción	0	ReturnStatus no presente
	FILL BIT STRING (SIZE(1))	0	Llenar con 0.
11	EID INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID = 0
12	ResponseParameter CONTAINER	**0**000 0010	Sin extensión, CHOICE 2_{10} = OCTET STRING
13		**0**000 0000	Sin extensión. Longitud del “string” = 0 bytes
	}		
14	FCS	xxxx xxxx	Secuencia verificadora de la trama
15		xxxx xxxx	
16	FLAG	0111 1110	“Flag” de término

C.5.3 CIERRE: RELEASE (UI)

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
1	FLAG	0111 1110	“Flag” inicial
2	Private LID	xxxx xxx**0**	Direccionamiento del enlace: con un OBE específico
3		xxxx xxx**0**	
4		xxxx xxx**0**	
5		xxxx xxx**1**	
6	MAC control field. L	1	La trama contiene un LPDU
	MAC control field. D	0	Dirección es “Down Link”
	MAC control field. A	0	RSE no asigna ventana privada en el “Up Link”
	MAC control field. C/R	0	LPDU tipo comando
	MAC control field. S	S	Bit de secuencia
	MAC control field. reserved bits	000	Bits reservados

Byte #	Atributo / Campo	Bits b_7 b_0	Descripción
7	LLC control field. n	N	Bit n de comando
	LLC control field. M	00	Comando UI
	LLC control field. P/F	0	No 1 = Poll, 0 = no Poll.
	LLC control field. M	00	
	LLC control field. reserved bits	11	No usados. Mantenerlos en 1.
8	Fragmentation header	1**fff f**001	Sin fragmentación. **ffff:** valor de número PDU incrementado secuencialmente.
9		0010	EVENT_REPORT.request
	RELEASE.request SEQUENCE		
	{		
	Indicador de Opción	0	Credencial de acceso no presente
	Indicador de Opción	0	EventParameter no presente
	Indicador de Opción	0	IID no presente
	Mode BOOLEAN	0	Modo no confirmado, no se espera respuesta
10	EID INTEGER (0..127,...)	**0**000 0000	Sin extensión, EID = 0 (Elemento de Sistema)
11	EventType INTEGER (0..127,...)	**0**000 0000	Sin extensión, RELEASE = 0.
	}		
12	FCS	xxxx xxxx	Secuencia verificadora de la trama
13		xxxx xxxx	
14	FLAG	0111 1110	"Flag" de término

Anexo D. Identificación de Concesiones

La tabla D1 entrega los números con que se identifican las concesiones del MOPTT.

TABLA D.1 Identificación de Concesiones MOPTT	
Identificador	***Concesión***
1	Sistema Oriente - Poniente
2	Sistema Norte - Sur
3	Sistema Américo Vespucio, Tramo Sur - Poniente
4	Sistema Américo Vespucio, Tramo Nor - Poniente
5	Acceso Vial Aeropuerto Arturo Merino Benítez
6	Autopista Acceso Nor - Oriente a Santiago
7	Variante Américo Vespucio - El Salto - Av. Kennedy
8	Autopista Santiago - San Antonio
9	Interconexión Vial Santiago - Valparaíso - Viña del Mar
10	Camino Santiago - Colina - Los Andes
11	Ruta 5, Tramo Los Vilos - La Serena
12	Ruta 5, Tramo Santiago - Los Vilos
13	Ruta 5, Tramo Santiago - Talca y Acceso Sur a Santiago
14	Ruta 5, Tramo Talca - Chillán
15	Ruta 5, Tramo Chillán - Collipulli
16	Ruta 5, Tramo Collipulli - Temuco
17	Ruta 5, Tramo Temuco - Río Bueno
18	Ruta 5, Tramo Río Bueno - Puerto Montt
19	Túnel El Melón
20	Camino Nogales - Puchuncaví
21	Red Vial Litoral Central
22	Variante Melipilla
23	Camino Internacional - Ruta 60
24	Camino de la Fruta - Ruta 66
25	Camino de Acceso Norte a Concepción
26	Camino de la Madera
27	Ruta Interportuaria Talcahuano - Penco
28	Aeropuerto Diego Aracena de Iquique
29	Aeropuerto Cerro Moreno de Antofagasta
30	Aeropuerto El Loa de Calama
31	Aeropuerto de Copiapó
32	Aeropuerto La Florida de La Serena

TABLA D.1 Identificación de Concesiones MOPTT	
Identificador	***Concesión***
33	Aeropuerto Puerto Velero de La Serena
34	Aeropuerto Internacional Arturo Merino Benítez
35	Aeropuerto Carriel Sur de Concepción
36	Aeropuerto de Temuco
37	Aeropuerto El Tepual de Puerto Montt
38	Aeropuerto Presidente Carlos Ibáñez del Campo de Punta Arenas
39	Puente Canal de Chacao
40	Coliseo Parque O'Higgins
41	Complejo Ecológico, Recreativo y Cultural Metropolitano
42	Recuperación Ribera Norte Río Bio Bio

Anexo E. Especificaciones Complementarias

A la fecha de la preparación de la presente versión, los estándares aplicables a la comunicación DSRC y a la transacción entre el punto de cobro y el transponder, dejan abiertas opciones que pueden conducir a soluciones no interoperables. Por este motivo, al menos transitoriamente, es necesario complementar los estándares con las especificaciones industriales definidas en el presente Anexo. En el futuro, y a medida que los estándares vayan incorporando las provisiones contenidas en estas especificaciones industriales que sean requeridas para la interoperabilidad, ellas perderán significación.

E.1 COMUNICACIONES DSRC

En materia de comunicaciones DSRC, la industria ha generado el documento:

Especificación Industrial: [GSS – 2.0]

Esta especificación será de cumplimiento obligatorio en los sistemas concesionados por el MOPTT. Entre otros, [GSS – 2.0] define un procedimiento uniforme de inicialización para establecer una comunicación interoperable, y fija los valores de un grupo de parámetros básicos del ambiente DSRC, todo al amparo de CEN TC278. Cabe señalar que una parte de las propuestas de [GSS – 2.0] ya ha sido incorporada a los estándares del CEN, lo que sumado a la difusión que ha adquirido en la industria, garantiza una plataforma estable para alcanzar la interoperabilidad entre equipos de diferentes fabricantes.

E.2 APLICACIÓN DE COBRO DE PEAJE

La industria ha propuesto un modelo de la Transacción de referencia entre el punto de cobro (RSE) y el transponder (OBE) en el documento:

Especificación Industrial: [A1]

[A1] debe ser considerado como un juego de herramientas para construir transacciones en un ambiente de cuentas centrales de clientes. En las obras concesionadas por el MOPTT, será obligatorio cumplir con los lineamientos fundamentales contenidos en [A1], y en particular con el marco detallado en E.2.1.

E.2.1 MARCO DEFINIDO POR [A1]

[A1] determina un conjunto de facilidades, que incluyen funciones, Atributos o datos, el mecanismo de seguridad, y la funcionalidad mínima del transponder, según se detalla a continuación:

- Funciones extraídas de [ISO – EFC]:
  - ✓ INITIALISATION

- ✓ GET
- ✓ SET
- ✓ ACTION
  - GET-NONCE
  - GET_STAMPED
  - SET_MMI
  - ECHO
  - GET_INSTANCE (optativo)
- ✓ EVENT REPORT
  - RELEASE

- Atributos o datos, extraídos de [ISO – EFC]:
  - ✓ EFC-ContextMark
  - ✓ ContractSerialNumber
  - ✓ ContractValidity
  - ✓ ReceiptServicePart
  - ✓ ReceiptFinancialPart
  - ✓ ReceiptAuthenticator
  - ✓ VehicleLicensePlateNumber
  - ✓ VehicleClass
  - ✓ VehicleDimensions
  - ✓ VehicleAxles
  - ✓ EquipmentStatus

- Otros Atributos, definidos en [CEN – L7], [GSS – 2.0] y [A1]:
  - ✓ ObeConfiguration
  - ✓ OBEGroupID
  - ✓ RndOBE

- Mecanismo de seguridad basado en el algoritmo DES, con claves diferenciadas para:
  - ✓ Acceso a la información guardada en el transponder
  - ✓ Autenticación de uno o más datos intercambiados en la Transacción
  - ✓ Autenticación de recibos grabados en el transponder

- Transponder provisto de interfaz de usuario o MMI, para entregar información al conductor.

Anexo F. Configuraciones Estandarizadas de Atributos Independientes de la Aplicación

Este anexo presenta las cuatro configuraciones de Atributos independientes de la aplicación o AIA estandarizadas por el MOPTT.

Un modelo de transponder real deberá tener todos los AIAs de una de las configuraciones definidas en el presente anexo, y cada AIA deberá tener las propiedades y ubicación aquí especificadas. Solamente con la aprobación previa del MOPTT se podrán incluir nuevos AIAs, o suprimir AIAs de una configuración determinada. Podrán existir AIAs adicionales de tipo privado, relativos a la programación y personalización del transponder, pero ellos no se usarán en las transacciones de las diferentes aplicaciones.

Las concesiones deberán registrar en el MOPTT la configuración usada en cada modelo de transponder que se distribuya. Para el proceso de homologación, el fabricante de transponder deberá indicar a la entidad homologadora la configuración usada en su producto.

F.1 CONFIGURACIÓN 1 DE AIAS

En la configuración 1, todos los AIAs se encuentran ubicados en el Elemento de Sistema. Los Atributos de este elemento se presentan en la Tabla F.1.

TABLA F.1 Atributos Independientes de la Aplicación de la Configuración 1

Nombre del Atributo	***AttrID*** *(1)*	***Longitud (Bytes)***	***Acceso*** *(2)*	***Comentarios***
ManufacturerID	1_{10}	2	RO	Emitido en la VST
ManufacturingSerialNumber	2_{10}	4	RO	
EquipmentClass	3_{10}	2	RO	Emitido en la VST
ActivityTimer	7_{10}	4	RO	
obeStatus	10_{10}	2	R/W	Emitido en la VST
BatteryInsertionDate	16_{10}	2	RO	
OBEGroupID	17_{10}	2	RO	Emitido en la VST
ElementAccessKey	120_{10}	8 (3)	NA	

1 AttrID: Número identificador del Atributo

2 NDA: Sin acceso directo RO: Sólo lectura ROnAC: Sólo lectura, no requiere credenciales de acceso R/W: Lectura y escritura NA: Sin acceso

3 Los algoritmos de seguridad en transponders según [A1] son de tipo DES y por lo tanto las claves respectivas son de 8 bytes. En el caso de que el transponder emplee algoritmos 3-DES, las claves serán de 16 bytes de longitud; para mantener compatibilidad con la especificación [A1], las claves deberán definirse con las mitades izquierda y derecha iguales.

F.1.1 ACCESO A LOS AIAS

Para la lectura de los AIAs es necesario presentar una credencial de acceso al elemento de sistema, el que tiene EID = 0. Dicha credencial se determina como se describe en 6.2, usando el valor de RndOBE emitido en la VST.

F.1.2 RESET DEL BIT T DE OBESTATUS

Para aplicar un reset al bit T o de manipulación ilegal del OBE, se escribe el valor 0 en el bit correspondiente del Atributo obeStatus.

F.2 CONFIGURACIÓN 2 DE AIAS

En la configuración 2, todos los AIAs se encuentran ubicados en el Elemento de Sistema. Los Atributos de este elemento se presentan en la Tabla F.2, la que para todos los efectos es igual a la Tabla F.1.

TABLA F.2 Atributos Independientes de la Aplicación de la Configuración 2

Nombre del Atributo	***AttrID*** *(1)*	***Longitud (Bytes)***	***Acceso*** *(2)*	***Comentarios***
ManufacturerID	1_{10}	2	RO	Emitido en la VST
ManufacturerSerialNumber	2_{10}	4	RO	
EquipmentClass	3_{10}	2	RO	Emitido en la VST
ActivityTimer	7_{10}	4	RO	
obeStatus	10_{10}	2	R/W	Emitido en la VST
BatteryInsertionDate	16_{10}	2	RO	
OBEGroupID	17_{10}	2	RO	Emitido en la VST
ElementAccessKey	120_{10}	8 (3)	NA	

Ver notas (1) a (3) al pie de la página 127

F.2.1 ACCESO A LOS AIAS

Para la lectura de los AIAs es necesario presentar una credencial de acceso al elemento de sistema, el que tiene EID = 0. Dicha credencial se determina en forma análoga a lo descrito en 6.2, pero usando en lugar del valor de RndOBE emitido en la VST, un nuevo valor aleatorio solicitado previamente al transponder, mediante el comando GetNonce para EID = 0.

F.2.2 RESET DEL BIT T DE OBESTATUS

Para aplicar un reset al bit T o de manipulación ilegal del OBE, se escribe el valor 0 en el bit correspondiente de obeStatus.

F.3 CONFIGURACIÓN 3 DE AIAS

En la configuración 3, todos los AIAs se encuentran ubicados en el Elemento de Sistema. Los Atributos de este elemento se presentan en la Tabla F.3.

TABLA F.3 Atributos Independientes de la Aplicación de la Configuración 3				
Nombre del Atributo	***AttrID*** *(1)*	***Longitud (Bytes)***	***Acceso*** *(2)*	***Comentarios***
ManufacturerID	1_{10}	2	RO	Emitido en la VST
ManufacturerSerialNumber	2_{10}	4	RO	
EquipmentClass	3_{10}	2	RO	Emitido en la VST
ActivityTimer	7_{10}	4	RO	
obeStatus	10_{10}	2	R/W	Emitido en la VST
BatteryInsertionDate	16_{10}	2	RO	
OBEGroupID	17_{10}	2	RO	Emitido en la VST
NumberOfWake-ups	18_{10}	2	RO	
NumberOfReleases	19_{10}	2	RO	
NumberOfVSTs	20_{10}	2	RO	
ElementAccessKey	120_{10}	8 (3)	NA	

Ver notas (1) a (3) al pie de la página 127

F.3.1 ACCESO A LOS AIAS

Para la lectura de los AIAs es necesario presentar una credencial de acceso al elemento de sistema, el que tiene EID = 0. Dicha credencial se determina como se describe en 6.2, usando el valor de RndOBE emitido en la VST.

F.3.2 RESET DEL BIT T DE OBESTATUS

Para aplicar un reset al bit T o de manipulación ilegal del OBE, se escribe el valor 0 en el bit correspondiente de obeStatus.

F.4 CONFIGURACIÓN 4 DE AIAS

La Tabla F.4 presenta los AIAs de la configuración 4. Se aprecia una implementación diferente a la de las configuraciones restantes.

TABLA F.4 Atributos Independientes de la Aplicación de la Configuración 4				
Nombre del Atributo	***AttrID (1)***	***Longitud (Bytes)***	***Acceso (2)***	***Comentarios***
ManufacturerID	-	2	NDA	Emitido en la VST
125 (Privado)	125_{10}	6	ROnAC	Ubicado en elemento con EID=2
TransponderSerialNumber		4/6	ROnAC	Parte del Atributo 125
BatteryInsertionDate		2/6	ROnAC	Parte del Atributo 125
EquipmentClass	-	2	NDA	Emitido en la VST
ActivityTimer	-	6	RO	Atributo leído mediante un comando privado
obeStatus	-	2	NDA	Emitido en la VST
OBEGroupID	37_{10}	2	ROnAC	Emitido en la VST. Este atributo está disponible en AID1/EID1, AID1/EID2, AID6/EID3 y AID29/EID4.

Ver notas (1) y (2) al pie de la página 127

F.4.1 ACCESO A LOS AIAS

Los Atributos con condición de acceso NDA no están disponibles para lectura directa. Ellos son emitidos en la VST.

La lectura del Atributo 125_{10} se efectúa mediante un comando GET.Request o GET_STAMPED.Request, sin necesidad de presentar credenciales de acceso.

La lectura del ActivityTimer se lleva a cabo mediante un comando privado, cuya codificación se entrega en C.2.5, y la correspondiente respuesta en C.2.6 y C.2.7.

F.4.2 RESET DEL BIT T DE OBESTATUS

Para aplicar un reset al bit T o de manipulación ilegal del OBE, se emplea un comando privado, cuya codificación se entrega en C.2.8, y la correspondiente respuesta en C.2.9 y C.2.10. Para esta operación, la clave usada en la determinación de la credencial de acceso se denomina TampK.