附录六：电子收费设备技术要求技术要求

目　录

附录六：电子收费设备技术要求

1 总则

（1）为了规范国内电子收费系统的联网应用，提高电子收费系统设计、建设及运营管理水平，确保电子收费技术能更好的服务于收费公路，特制定本技术要求。

（2）本技术要求给出了在国内公路收费领域内使用的电子收费设备的关键技术参数及基本功能、性能要求。这些指标要求既作为电子收费设备的市场准入基本条件，又作为交通行业管理部门、收费公路经营管理者等进行电子收费设备选型的依据。

（3）本技术要求适用于电子收费系统的新建及改建工程。

（4）本技术要求未涉及到的内容，可参照现行有关国内、国际标准。

2 名词与定义

2.1 唤醒 wakeup

OBU 由休眠状态转换为工作状态的过程。

2.2 前导码 preamble

物理层帧信息的前置信号，与链路层无关，可以是调制或者未调制的载波。

2.3 应用 application

DSRC 协议服务的用户。

2.4 文件 file

车载单元（OBU）应用数据的基础组织单位，一般多个相关的数据单元组成一个文件。

2.5 文件标识 file identifier

文件的标识号码。同一目录下，文件标识号是唯一的。

2.6 目录标识 directory identifier

明确识别某目录的标志。

2.7 广播 broadcast

路侧单元（RSU）以广播地址发出信息，面向所有 OBU 且不需要 OBU 回复

的通信应用。

2.8 初始化 initialization

RSU 发起的与 OBU 之间协商彼此通信参数配置和应用的过程。

2.9 层管理 layer management

提供 DSRC 通信参数的值以及采集和发布其他控制通信系统所必需的信息，用于支持通信系统的管理。

2.10 配置 profile

有关功能、性能、不同层的设置或应用处理的信息。用一个整型数值来标识。

3 符号和缩略语

3.1 符号

下列符号适用于本技术要求。
dBm 表征功率与1mW的比值，0dBm=1mW
RSUt 路侧单元发射天线
RSUr 路侧单元接收天线
V（RB） 接收成败状态变量
V（RI） 接收序列状态变量
V（SI） 发送序列状态变量

3.2 缩略语

下列缩略语适用于本技术要求。
ACK 确认（Acknowledge）
ACn 有确认的命令/响应（Acknowledged Command/Response）
ADU 应用数据单元（Application Data Unit）
AID 应用标识（Application Identifier）
APDU 应用协议数据单元（Application Protocol Data Unit）
ASDU 应用服务数据单元（Application Service Data Unit）
ASK 幅移键控（Amplitude Shift Keying）
ASN.1 抽象语法记法一（Abstract Syntax Notation One）
AVI 自动车辆识别（Automatic Vehicle Identification）
BER 位误码率（Bit Error Rate）
B-KE 广播内核（Broadcast Kernel）
BST 信标服务表（Beacon Service Table）
C/R 命令/响应（Command/Response）
DID 目录标识（Directory Identifier）
DSRC 专用短程通信（Dedicated Short Range Communication）

e.i.r.p　等效全向辐射功率（Equivalent Isotropically Radiated Power）
ETC　电子收费（Electronic Toll Collection）
F　结束（Final）
FCS　帧校验序列（Frame Check Sequence）
FID　文件标识（File Identifier）
FSK　频移键控（Frequency Shift Keying）
ICC　集成电路卡（Integrate Circuit Card）
IID　启用标识（Invoker Identifier）
I-KE　初始化内核（Initialization Kernel）
LLC　逻辑链路控制（Logical Link Control）
LPDU　逻辑链路控制协议数据单元（LLC Protocol Data Unit）
LSAP　逻辑链路控制服务访问点（LLC Service Access Point）
LSB　最低有效位（Least Significant Bit）
LSDU　逻辑链路控制服务数据单元（LLC Service Data Unit）
LID　链路标识（Link Identifier）
L1　DSRC 物理层　（Layer1）
L2　DSRC 数据链路层（Layer2）
L7　DSRC 应用层（Layer7）
M　修改功能位（Modifier function bit）
MAC　媒体访问控制（Medium Access Control）
MAC　信息鉴别码（Message Authentication Code）
MTC　人工半自动收费（Manual Toll Collection）
MSB　最高有效位（Most Significant Bit）
OBE　车载设备（On Board Equipment）
OBU　车载单元（On Board Unit）
PDU　协议数据单元（Protocol Data Unit）
PER　紧缩编码规则（Packed Encoding Rules）
P/F　询问/终止（Poll/Final）
PPDU　物理层协议数据单元（Physical layer Protocol Data Unit）
PSAM　消费安全访问模块（Payment Security Access Module）
R　响应（Response）
RID　记录标识（Record Identifier）
RSE　路侧设备（Roadside Equipment）
RSU　路侧单元（Roadside Unit）
SAP　服务访问点（Service Access Point）
SDU　服务数据单元（Service Data Unit）
TDES　三重数据加密标准（Triple Data Encryption Standard）
T-APDU　传送-应用协议数据单元（Transfer Application Protocol Data Unit）
T-ASDU　传送-应用服务数据单元（Transfer Application Protocol Data Unit）
T-KE　传送内核（Transfer Kernel）
XPD　交叉极化鉴别率（Cross Polarization Discrimination）
UI　无编号信息（Unnumbered information）
VST　车辆服务表（Vehicle Service Table）

4 物理层

物理层包括下、上行链路的要求。

载波频率、e.i.r.p 和杂散发射的要求应符合信部无[2002]277 号文《关于使用 5.8GHz 频段频率事宜的通知》的规定。

4.1 下行链路技术要求

下行链路参数 **表 1**

序号	参数		限值
1	载波频率	信道1	5.830GHz
		信道2	5.840GHz
2	占用带宽		≤5MHz
3	频率容限		±10×10^{-6}
4	e.i.r.p		≤+33dBm
5	杂散发射	30MHz～1000MHz	≤-36dBm／100kHz
		2400MHz～2483.5MHz	≤-40dBm／1MHz
		3400MHz～3530MHz	≤-40dBm／1MHz
		5725MHz～5850MHz[a]	≤-33dBm／100kHz
		其它1GHz～20GHz	≤-30dBm／1MHz
6	邻道泄漏功率比		-30dB
7	天线半功率波瓣宽度	水平面	＜38°
		垂直面	＜45°
8	天线极化		右旋圆极化
9	XPD	最大增益方向	RSUt≥15dB
		-3dB区域	RSUt≥10dB
10	调制方式		ASK
11	调制系数		调制系数：0.5～0.9
12	编码方式		FM0
13	位速率		256kbps
14	位时钟精度		±100×10^{-6}
15	OBU唤醒方式		15～17个周期14kHz方波[b]
16	OBU唤醒时间		＜5ms
17	OBU唤醒灵敏度		≤-40dBm
18	OBU接收灵敏度		≤-50dBm
19	OBU接收带宽		5.825GHz～5.845GHz
20	BER		10×10^{-6}以内
21	前导码		16位“0”

a 对应载波2.5倍信道带宽以外。

b RSU强制要求发送该波形；OBU可选择被该波形唤醒或者被正常通信帧信号唤醒。

4.2 上行链路技术要求

上行链路参数　　表 2

序号	参数		限值
1	载波频率	信道1	5.790GHz
		信道2	5.800GHz
2	占用带宽		≤5MHz
3	频率容限		±200×10^{-6}
4	e.i.r.p		≤+10dBm
5	杂散发射	30MHz～1000MHz	≤-36dBm／100kHz
		2400MHz～2483.5MHz	≤-40dBm／1MHz
		3400MHz～3530MHz	≤-40dBm／1MHz
		5725MHz～5850MHz	≤-33dBm／100kHz
		其它1GHz～20 GHz	≤-30dBm／1MHz
6	邻道泄漏功率比		30dB
7	天线半功率波瓣宽度		＜70°
8	天线极化		线极化或右旋圆极化
9	XPD	最大增益方向	RSUr≥15dB
		-3dB区域	RSUr≥10dB
10	调制方式		ASK
11	调制系数		调制系数：0.5～0.9
12	编码方式		FM0
13	位速率		512kbps
14	位时钟精度		±100×10^{-6}
15	RSU接收灵敏度		≤-70dBm
16	BER		10×10^{-6}以内
17	前导码		16位“0”
a 对应载波2.5倍信道带宽以外。			

5 数据链路层

5.1 链路层主要参数

链路层主要参数见表 3。

链路层主要参数　　表 3

参数	参数定义	参数取值
T1	下行链路帧与后面相邻的上行链路帧的最短间隔时间	160μs
T2	上行链路帧与后面相邻下行链路帧的最短间隔时间	32μs
T3	下行链路帧与后面相邻下行链路帧的最短间隔时间	10μs
Tu	时间单位	448μs
N1	专用链路建立请求延时计数器	0～7
N2	第二层帧的最长八位位组数	128
N3	内部传送计数器	—

N4	确认定时器	—
注：本技术要求对N3、N4的参数取值不做规定。		

5.2 信息帧

5.2.1 信息帧格式

数据链路层信息交互应以帧形式进行，帧包含 MAC 地址、MAC 控制域、LPDU（可选）和帧校验几部分。根据 LPDU 的类型又分为含命令 LPDU 的帧和含响应 LPDU 的帧，分别见图 1 和图 2。

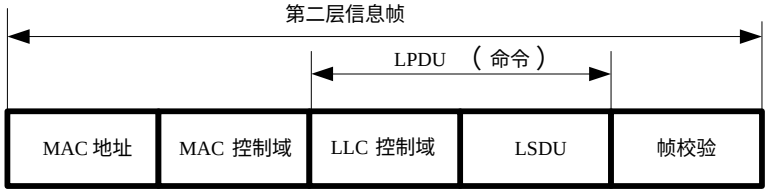


图1 含命令 LPDU 的信息帧结构

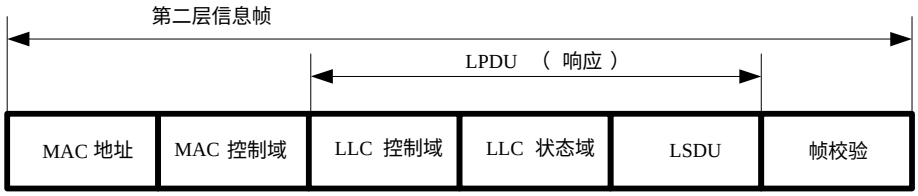


图2 含响应 LPDU 的信息帧结构

不包含 LPDU 域的帧结构见图 3。

图3 不包含 LPDU 域的帧结构

5.2.2 帧封装方式

5.2.2.1帧封装格式

信息帧采用同步传输方式，帧封装格式见图 4。

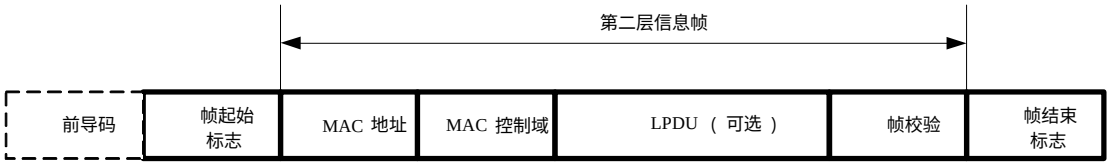


图4 第二层信息帧封装格式

5.2.2.2帧起始标志和帧结束标志

帧起始标志和帧结束标志格式均为二进制序列 0111 1110。

前一帧的帧结束标志不能作为后一帧的帧起始标志。

如果接收端收到多个连续的帧起始标志，则以最后一个作为帧的起始。

5.2.2.3前导码

前导码的规定见 4.1 和 4.2。

5.2.2.4透明传输

帧起始标志和帧结束标志间的信息，不含帧起始标志和帧结束标志，都应进行插 0 处理，过程如下。

发送端连续发送五个“ 1 ”后插入一个“ 0 ”。

接收端连续收到五个“ 1 ”时检查第六个比特。如果第六个比特为“ 0 ”，则删除该“ 0 ”；如果为“ 1 ”，则检查第七个比特。如果第七个比特为“ 0 ”，表示起始或者结束标志；如果为“ 1 ”，则接收端视该信息帧为无效帧，并舍弃。

5.2.2.5MAC 地址

媒体访问地址分为广播 MAC 地址和专用 MAC 地址。广播 MAC 地址用于 RSU 对所有 OBU 的访问；专用 MAC 地址用于对某特定 OBU 的访问。

广播 MAC 地址取值应为 32 位全“ 1 ”比特：0xFFFFFFFF；专用 MAC 地址取值应为 32 位非全“ 1 ”比特。

MAC 地址采用固定分配方式，由交通行业技术主管部门统一管理。

5.2.3 MAC 控制域

5.2.3.1下行链路的 MAC 控制域

下行链路的 MAC 控制域由 RSU 发送的帧使用。格式见表 4。

下行链路 MAC 控制域 **表 4**

比特位	标识符	含义	取值
7	D/U	方向标识符	0：下行链路
6	L	LPDU是否存在	1：存在；0：不存在
5	C/R	命令/响应	0：命令
4	Q	广播信息，只有MAC地址为全1才有效	0：不寻求建立专用链路； 1：寻求建立专用链路
3—0	—	保留	—

5.2.3.2上行链路的 MAC 控制域

上行链路的 MAC 控制域由 OBU 发送的帧使用。格式见表 5。

上行链路 MAC 控制域 **表 5**

比特位	标识符	含义	取值
7	D/U	方向标识符	1：上行链路
6	L	LPDU是否存在	1：存在；0：不存在
5	C/R	命令/响应	1：响应

4－0	—	保留	—

5.2.4 LPDU 格式

LPDU 格式见 5.5.3.1。

5.2.5 帧校验序列

帧结束标志前应有 16 比特的 FCS。FCS 的计算范围包括 MAC 地址、MAC 控制域和 LPDU。

FCS 应符合 GB/T 7496 中定义的 16 比特帧校验序列。生成多项式为 $X^{16}+X^{12}+X^5+1$，使用的初始值是 0xFFFF。

5.2.6 比特顺序

帧起始标志、帧结束标志、MAC 地址、MAC 控制域和 LPDU 应先传送 LSB。FCS 应从 MSB 开始传送。

5.3 专用通信链路建立与撤销

5.3.1 专用通信链路建立

RSU 与 OBU 之间的通信支持广播和点对点两种方式。

广播方式下，RSU 与 OBU 之间不需要建立专用通信链路，以广播 MAC 地址作为链路标识，所有 OBU 都能接收 RSU 发出的信息。

点对点方式下，RSU 与 OBU 之间需建立专用通信链路，该链路以专用 MAC 地址作为唯一标识。专用链路的建立过程如下：

（1） RSU 周期性广播 Q 为 1 的特定信息；

（2） 通信区域内 OBU 收到该信息后，随机延时 N1 个时间单位 Tu；

（3） OBU 发送包括其 MAC 地址的信息到 RSU；

（4） RSU 确认收到合法帧后，登记对应的 OBU MAC 地址，并以该 MAC 地址与对应的 OBU 通信；

（5） OBU 收到带本 OBU MAC 地址的下行链路帧后，专用链路建立成功。

5.3.2 专用通信链路撤销

专用通信链路的撤销以及 RSU 对 OBU MAC 地址的注销，由 RSU 逻辑自行确定。

5.4 MAC 子层

5.4.1 MAC 服务原语

MAC 子层面向 LLC 子层提供以下服务原语：

（1）MAC.request：LLC 子层发送给 MAC 子层，请求发送一个 LPDU；

（2）MAC.indication：MAC 子层发送到 LLC 子层，指示成功收到一个 LPDU。

RSU 与 OBU 中的 MAC 子层服务原语分别为 B-MAC 原语与 M-MAC 原语，

见表 6。

MAC 服务原语 **表 6**

服务原语	条目	描述
B-MAC.request	原型	B-MAC.request(MAC 地址，LPDU)
	功能	LLC 子层发送到 MAC 子层，请求发送一个 LPDU 给 OBU
	参数	MAC 地址：要发送给 OBU 的 MAC 地址，它可以是专用 MAC 地址、广播 MAC 地址； LPDU：链路协议数据单元
B-MAC.indication	原型	B-MAC.indication(MAC 地址，LPDU)
	功能	MAC 子层发送到 LLC 子层，指示成功接收到一个有效帧
	参数	MAC 地址：接收到的帧中 MAC 地址域的内容； LPDU：链路协议数据单元
M-MAC.request	原型	M-MAC.request(MAC 地址，LPDU)
	功能	LLC 子层发送到 MAC 子层以请求发送一个 LPDU 给 RSU
	参数	MAC 地址：OBU 的专用 MAC 地址； LPDU：链路协议数据单元
M-MAC.indication	原型	M-MAC.indication（ MAC 地址，LPDU）
	功能	MAC 子层发送到 LLC 子层，指示成功接收到一个有效帧
	参数	MAC 地址：是接收到的帧中 MAC 地址域的内容； LPDU：链路协议数据单元

5.4.2 MAC 服务描述

5.4.2.1路侧单元 MAC 服务

（1）专用链路建立

RSU 广播 Q 等于 1 的特定 BST 信息，OBU 以专用 MAC 地址响应该 BST 信息。

RSU 确认收到合法帧后，登记 OBU 的 MAC 地址，并采用该 MAC 地址与 OBU 通信；OBU 端的处理过程见 5.4.2.2 的（1）。

（2）帧接收

① 检测帧的有效性

MAC 子层应检查所有接收到的帧的有效性，依据条件如下：

(a) 帧起始标志和帧结束标志符合 5.2.2.2 的规定；

(b) 去掉为保持透明性而插入的 0 比特后，该帧包含的比特数应为 8 的整数倍，且不多于 N2 个八位位组；

(c) 包含一个有效的专用 MAC 地址，如果已经建立专用链路，则此 MAC 地址应与专用链路的 MAC 地址相符；

(d) 包含一个符合要求的 MAC 控制域；

(e) 包含一个有效的 FCS。

② 信息接收

如果接收到的有效帧的 L 比特为 1，表明该帧包含一个 LPDU。从帧中提取 LPDU 和 MAC 地址域的内容，以 B-MAC.indication 形式传送给 LLC 子层。

如果接收到的有效帧的 L 比特为 0，表明该帧不包含 LPDU。

（3）帧发送

RSU MAC 接收 LLC 提供的 LPDU，并根据帧格式构建一个帧，MAC 控制域的 L 比特应置 1，D 比特应置 0，如果为广播信息则根据是否要寻求建立专用链路设置 Q。然后将该帧传送给低层。

5.4.2.2 OBU MAC 服务

（1）专用链路建立

OBU 接收到发自广播 MAC 地址且 Q 等于 1 的广播后，以专用 MAC 地址向 RSU 发送信息。

OBU 收到第一个与自身 MAC 地址相符的下行帧后表示专用链路建立。

（2）帧接收

① 检测帧的有效性

MAC 子层应检查所有接收到的帧的有效性：

(a) 帧起始标志和帧结束标志符合 5.2.2.2 的规定；

(b) 去掉为保持透明性而插入的 0 比特后，该帧包含的比特数应为 8 的整数倍，且不多于 N2 个八位位组；

(c) 包含一个有效地址域指示合法 MAC 地址；

(d) 包含一个符合要求的 MAC 控制域；

(e) 包含一个有效的 FCS。

②信息接收

如果接收到的有效帧的 L 比特为 1，表明该帧包含一个 LPDU。从帧中提取 LPDU 和 MAC 地址域的内容，以 M-MAC.indication 形式传送给 LLC 子层。

（3）帧发送

OBU MAC 接收 LLC 提供的 LPDU，并根据帧格式构建一个帧，MAC 控制域的 L 比特和 D 比特应置 1。然后将该帧传送给低层。

5.5 LLC 子层

5.5.1 总则

LLC 产生用于传输的命令 PDU 和响应 PDU，并解释接收的命令 PDU 和响应 PDU。LLC 规定的功能包括：

（1） 控制信息的初始化；

（2） 组织数据流；

（3） 解释接收到的命令 PDU 并生成适当的响应 PDU；

（4） LLC 子层的差错控制与差错恢复。

LLC 子层规定对等实体间信息和控制传输的协议进程，其逻辑链路控制操作包括两种类型。

类型 1 操作规定一个具有最小协议复杂度的不确认无连接方式的服务。在上层提供了基本数据恢复和顺序功能时使用此类型操作。

类型 3 操作规定一个确认无连接方式的数据单元交换服务，它允许一个站点在传送数据的同时又请求回传数据。

5.5.2 LLC 子层服务规范

5.5.2.1总则

LLC 子层规定 LLC 子层用户对 LLC 子层所要求的服务，这些服务使 LLC 子层用户可利用 LLC 子层进行数据包交换。

LLC 子层提供两种服务方式：不确认无连接方式和确认无连接方式。

不确认无连接方式：该数据传输服务提供一组方法，使数据链路用户实体可采取不确认的方式交换 LSDU，而无需在数据链路层上建立连接。该数据传输可以是点对点、组播或广播。

确认无连接方式：该数据单元交换服务提供一组方法，使数据链路用户实体可以在不建立数据链路连接的情况下交换 LSDU，并在 LLC 子层进行确认。该数据交换是点对点的。

5.5.2.2交互过程概述

（1）不确认无连接方式服务

与不确认无连接方式数据传送有关的原语是：

DL-UNITDATA.request

DL-UNITDATA.indication

DL-UNITDATA.request 从 LLC 子层用户传递给 LLC 子层，请求使用不确认无连接方式发送一个 LSDU。

DL-UNITDATA.indication 从 LLC 子层传递给 LLC 子层用户，指示一个 LSDU 的到达。

（2）确认无连接方式服务

①确认无连接方式数据传送

与确认无连接方式数据单元传送服务相关的原语是：

DL-DATA-ACK.request

DL-DATA-ACK.indication

DL-DATA-ACK-STATUS.indication

DL-DATA-ACK.request 从 LLC 子层用户传递给 LLC 子层，请求使用确认无连接方式数据单元传送过程发送一个 LSDU。

DL-DATA-ACK.indication 从 LLC 子层传递给 LLC 子层用户，指示一个命令 PDU 的到达，该 PDU 仅被用作再同步时除外。

DL-DATA-ACK-STATUS.indication 从 LLC 子层传递给 LLC 子层用户，传达之前与其对应的 DL-DATA-ACK.request 的执行结果。

②确认无连接方式数据交换

与确认无连接方式数据单元交换服务相关的原语是：

DL-REPLY.request

DL-REPLY.indication

DL-REPLY-STATUS.indication

DL-REPLY.request 从 LLC 子层用户传递给 LLC 子层，请求使用确认无连接方式数据单元交换过程从一个远端站点返回一个 LSDU 或在站点之间交换 LSDU。

DL-REPLY.indication 从 LLC 子层传递给 LLC 子层用户，指示一个命令 PDU 的到达。

DL-REPLY-STATUS.indication 从 LLC 子层传递给 LLC 子层用户，传达之前与其对应的 DL-REPLY.request 的执行结果。

③确认无连接方式待传数据更新

与确认无连接方式待传数据更新服务相关的原语是：

DL-REPLY-UPDATE.request

DL-REPLY-UPDATE-STATUS.indication

DL-REPLY-UPDATE.request 从 LLC 子层用户传递给 LLC 子层，请求 LLC 子层保存一个 LSDU，并且在稍后其他站点请求 LSDU 时发送。

DL-REPLY-UPDATE-STATUS.indication 从 LLC 子层传递给 LLC 子层用户，传达之前与其对应的 DL-REPLY-UPDATE.request 的执行结果。

5.5.2.3详细服务规范

（1）总则

详细服务规范详细规定 LLC 的服务原语及其参数，显示 LLC 与上层协议层间信息传送关系的逻辑序列图见图 5。

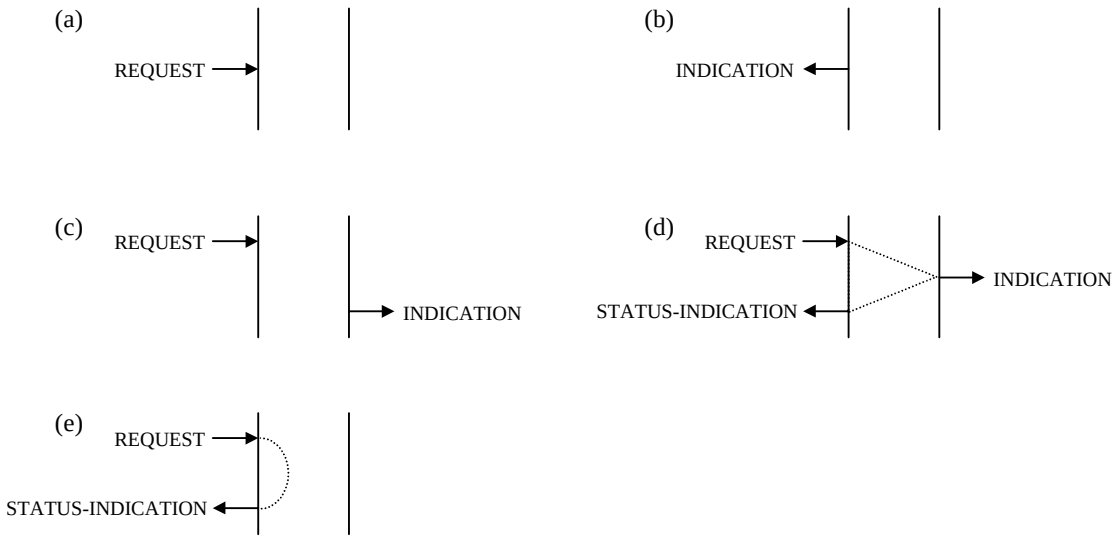


图5　逻辑序列图

（2）DL-UNITDATA.request

不确认无连接方式数据传送服务的服务请求原语。

OBU 的 DL-UNITDATA.request 应提供如下参数：

DL-UNITDATA.request（
LID，
data
）

LID 应指定一个专用 MAC 地址。

RSU 的 DL-UNITDATA.request 应提供如下参数：

DL-UNITDATA.request（
LID，

data，
response request
）

LID 可以指定一个专用或广播 MAC 地址。

“ response request ”表示是否要求 OBU 立即响应，若要求立即响应，则 RSU 随后等待应答。“ response request ”参数被直接传递给 MAC 子层。MAC 子层根据“ response request ”参数设置 Q 比特。

该原语从 LLC 子层用户传递给 LLC 子层，请求采用不确认无连接方式发送一个 LSDU。

（3）DL-UNITDATA.indication

不确认无连接方式数据传送服务的服务指示原语。

该原语应提供如下参数：

DL-UNITDATA.indication（
LID，
data
）

LID 在 OBU 端应指定专用 MAC 地址，在 RSU 端可指定专用 MAC 地址或广播 MAC 地址。

该原语从 LLC 子层传递到 LLC 子层用户，指示一个 LSDU 的到达。

（4）DL-DATA-ACK.request

确认无连接方式数据传送服务的服务请求原语。

该原语应提供如下参数：

DL-DATA-ACK.request（
LID，
data
）

LID 应指定专用 MAC 地址。

该原语从 LLC 子层用户传递到 LLC 子层，请求使用确认无连接方式传送一个 LSDU。

（5）DL-DATA-ACK.indication

确认无连接方式数据传送服务的服务指示原语。

该原语应提供如下参数：

DL-DATA-ACK.indication（
LID，
data
）

LID 应指定一个专用 MAC 地址。

该原语从 LLC 子层传递给 LLC 子层用户，用来指示一个非空、非重复 LSDU 的到达。

（6）DL-DATA-ACK-STATUS.indication

确认无连接方式数据传送服务的服务状态指示原语。

该原语应提供如下参数：

DL-DATA-ACK-STATUS.indication（

LID，
status
）

LID 应指定专用 MAC 地址。

“ status ”指示之前与其对应的数据传送服务请求成功还是失败。

该原语从 LLC 子层传递给 LLC 子层用户，指示之前与其对应的确认无连接方式数据单元传送服务请求成功还是失败。

（7）DL-REPLY.request

确认无连接方式数据交换服务的服务请求原语。

该原语应提供如下参数：

DL-REPLY.request（
LID，
data
）

LID 应指定专用 MAC 地址。

该原语从 LLC 子层用户传递给 LLC 子层，请求使用确认无连接方式数据交换过程向远端站点请求预先准备好的 LSDU，或与远端站点交换 LSDU。

（8）DL-REPLY.indication

确认无连接方式数据交换服务的服务指示原语。

该原语应提供如下参数：

DL-REPLY.indication（
LID，
data
）

LID 应指定专用 MAC 地址。

该原语从 LLC 子层传递给 LLC 子层用户，指示成功的从远端站点接收到一个对 LSDU 的请求，或指示其与远端站点 LSDU 的交换。

向请求站点传送预先准备好的 LSDU 不应破坏该 LSDU 的原始拷贝。后续任何站点发来的数据请求都会以相同的 LSDU 作为响应，直到使用 DL-REPLY-UPDATE.request 用新的信息替换该 LSDU。

（9）DL-REPLY-STATUS.indication

确认无连接方式数据交换服务的服务状态指示原语。

该原语应提供如下参数：

DL-REPLY-STATUS.indication（
LID，
data，
status
）

LID 应指定专用 MAC 地址。

“ status ”指示之前与其对应的确认无连接方式数据交换请求是成功还是失败。

该原语从 LLC 子层传递给 LLC 子层用户，指示之前与其对应的确认无连接方式数据交换请求是成功还是失败，并将 LSDU 传递给 LLC 子层用户。

（10）DL-REPLY-UPDATE.request

应答数据单元准备服务的服务请求原语。

该原语应提供如下参数：

DL-REPLY-UPDATE.request（

LID，

data

）

LID 应指定专用 MAC 地址。

该原语从 LLC 子层用户传递给 LLC 子层，请求 LLC 子层保存一个 LSDU，用于稍后的传送请求。

（11）DL-REPLY-UPDATE-STATUS.indication

应答数据单元准备服务的服务确认原语。

该原语应提供如下参数：

DL-REPLY-STATUS.indication（

LID

status

）

LID 应指定专用 MAC 地址。

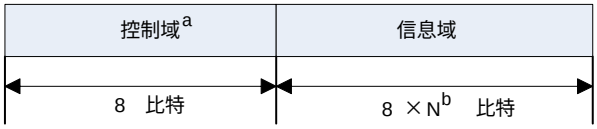
“status”指示先前相关的应答数据单元准备请求是成功还是失败。

该原语由 LLC 子层传递给 LLC 子层用户，指示之前与其对应的数据单元准备请求是成功还是失败。

5.5.3 LPDU 结构

5.5.3.1LPDU 格式

LPDU 格式见图 6。



a 控制域见 5.5.3.2.3。

b N 为八位位组的个数，是一个大于或等于 0 的整数。

图6 LPDU 格式

5.5.3.2LPDU 元素

（1）地址域

LLC子层和MAC子层使用相同的MAC地址。

下行链路中的 LID 应指示 LLC 的信息域将要发往的一个或多个目的站点。上行链路中的 LID 应指示发出 LLC 信息域的源站点。

MAC地址域的格式见5.2.2.5。

（2）C／R 比特

C／R 比特位于 MAC 控制域的第四比特。如果该比特为 0，表明该 LPDU 是一个命令；如果该比特为 1，表明该 LPDU 是一个响应。

（3）LLC 控制域

LLC 控制域应由一个八位位组构成，用来详细定义命令和响应的功能。LLC 控制域的内容见 5.5.5。

（4）信息域

信息域应由若干个（包括 0）八位位组构成。

（5）比特顺序

MAC 子层在发送／接收响应和命令时应从 LSB 开始。信息域在传递给 MAC 子层时的比特顺序应与其从第七层接收到的比特顺序相同；信息域在传递给第七层时的比特顺序也应和从 MAC 子层接收到的比特顺序相同。

（6）无效 LPDU

当符合下列情况之一时，LPDU 应定为无效。

①已被 MAC 子层识别出为无效；

②长度不是整数个八位位组；

③长度为 0(没有控制域)；

④没有包含有效的命令或响应控制域；

⑤包含类型 3 的命令或响应控制域，但 LID 是广播 MAC 地址；

⑥包含类型 3 的响应控制域，但在其信息域中没有 ACn 响应状态子域。

5.5.4 LLC 规定的两类操作

5.5.4.1类型 1 操作

通过类型 1 操作，无需建立数据链路连接，就能在 LLC 实体间交换 PDU。在 LLC 子层 PDU 无需被确认，也不应有任何的流控制和差错恢复功能。

5.5.4.2类型 3 操作

通过类型 3 操作，无需建立数据链路连接，就能在 LLC 实体间交换 PDU。在 LLC 子层 PDU 都应被确认。确认功能是通过从目的 LLC 返回给源 LLC 一个包含在独立 PDU 中的特定响应来完成的，该 PDU 包含状态信息并且可以包含或不包含用户信息。

在正常的操作中，每一个类型 3 操作的命令 PDU 都应收到一个确认 PDU。源 LLC 出于恢复目的可能会重传一个类型 3 的命令 PDU，但是对于同一个 MAC 地址，LLC 在等待先前的 PDU 的确认时不会传输一个新的类型 3 PDU。LLC 实体不会从第七层接受新的请求原语，直到其收到目的 LLC 实体对先前的“请求”原语的 LPDU 的确认。在 LLC 传输（重传之后）PDU 失败的情况下，这种约束使上层能在重新开始正常的数据传输之前执行恢复操作。

LLC 控制域代码交变机制为连续的 PDU 提供了一个 1 比特的序列号，使接收到命令 PDU 的 LLC 能区分该 PDU 是一个新的 PDU 还是先前接收到的 PDU 的副本。此外，接收确认 PDU 的 LLC 可以确信该确认信息是针对最近一次发送的 PDU 的。超时的确认信息应被忽略。

类型 3 操作定义了状态信息，该信息应由参与信息交换的站点进行维护。每个站点必须维护一个 1 比特序列码用于发送，一个 1 比特序列码用于接收。

类型 3 操作只用于点对点通信。

5.5.5 LLC 程序元素

5.5.5.1控制域格式

控制域格式见图 7。

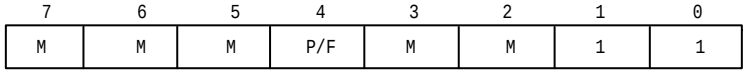


图7　LPDU 控制域

PDU 提供数据链路控制功能和信息传送。

PDU 应该包含一个依照 5.5.6.2 设定的 P／F 比特。

5.5.5.2控制域参数

（1）类型 3 操作参数

①V(SI)

发送类型 3 命令时，LLC 应维护一个 V（SI)。该变量被置成所收到的最后一个类型 3 响应 PDU 的控制域代码第八比特的值。V（SI）变量使 LLC 能确认其收到的确认对应于当前“ 尚未完成 ”的信息传输，同时使接收方能检测到重复的帧。

V（SI）应在建立一个新的 LID 时创建。

②V(RI)

发送类型 3 命令时，LLC 应维护一个 V（RI)。该变量包含的值与收到的最后一个类型 3 命令的 AC0 或 AC1 控制域代码第八比特相反。V（RI）使 LLC 可以区分所收到的类型 3 命令 PDU 是首次接收到，还是一个先前已收到的 PDU 的重传。

V（RI） 应在建立一个新的 LID 时创建。

③V(RB)

发送类型3命令时，LLC应维护一个V（RB）。V（RB）指示最后收到的类型3命令接收是成功还是失败。V（RB）确保对重复接收到的命令PDU的响应和对原始命令PDU的响应包含相同的接收状态。如果先前一次接收失败而最近一次接收成功，接收成败状态变量V（RB）应被改变。

5.5.5.3命令和响应

（1）概述

（2）和（3）分别说明了类型 1 和类型 3 操作的每一种有效的控制域设置所对应的命令和响应集。MAC 控制域中的第四比特 C／R 比特用于区分命令和响应。类型 1 和类型 3 操作的命令和响应见表 7。

类型 1 和类型 3 操作的命令和响应　表 7

命令	响应
UI 无编号信息	—
ACn，n=0－有确认无连接信息　序列 0	ACn，n=0－有确认无连接确认　序列 0
ACn，n=1－有确认无连接信息　序列 1	ACn，n=1－有确认无连接确认　序列 1

（2）类型 1 操作命令

类型 1 操作命令 PDU 的 LLC 控制域见图 8。

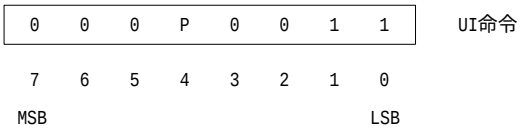


图8　类型 1 操作命令控制域比特分配

下行链路中，UI 命令 PDU 用于发送信息给一个或多个 OBU。上行链路中，UI 命令 PDU 用于向一个 RSU 发送信息。

UI 命令 PDU 可以在目的 LLC 和源 LLC 之间没有预先建立数据链路连接的情况下使用。对于 UI 命令 PDU，不存在 LLC 响应 PDU。

传送命令 PDU 的过程中，如果出现数据链路异常，包含在该 UI 命令 PDU 中的数据可能丢失。

（3）类型 3 操作命令和响应

①总则

类型 3 操作命令和响应 PDU 的 LLC 控制域见图 9。

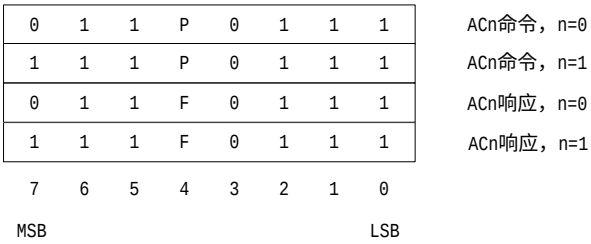


图9　类型 3 操作命令和响应的控制域

②ACn 命令

在类型 3 操作中，ACn 命令 PDU 可在没有预先建立数据链路连接的情况下用来发送或请求信息。ACn 命令 PDU 的使用不要求目的和源之间存在数据链路连接。接收到一个 ACn 命令 PDU 后应尽早以 ACn 响应 PDU 进行确认。ACn 命令应使用专用 MAC 地址。ACn 命令 PDU 的信息域可以为空或非空。若为非空，应包含一个 LSDU。

③ACn 响应

在类型 3 操作中，使用 ACn 响应 PDU 应答一个 ACn 命令 PDU。ACn 响应 PDU 应发送给源 LLC，并标识出进行应答的 LLC。ACn 响应 PDU 的信息域应包含一个状态子域（见（4））。

根据 P／F 比特状态和 LSDU 是否为空，ACn 命令 PDU 实现的功能见表 8。

ACn 命令 PDU 功能表　　表 8

P	LSDU	功能
0	空	再同步
0	非空	发送数据
1	空	请求数据
1	非空	交换数据

ACn 响应 PDU 实现的功能见表 9。

ACn 响应 PDU 功能表　　表 9

F	**LSDU**	**功能**
0	空	再同步的确认或对接收到数据的确认

0	非空	不允许
1	空	确认，请求的数据无法获得
1	非空	确认，携带了被请求的数据

（4）类型 3 操作的响应信息域

每个 ACn 响应 PDU 的信息域应包含一个状态子域，信息域的其它部分可以为空或非空。如果非空，则必须包含一个 LSDU。ACn 响应信息域格式见图 10。

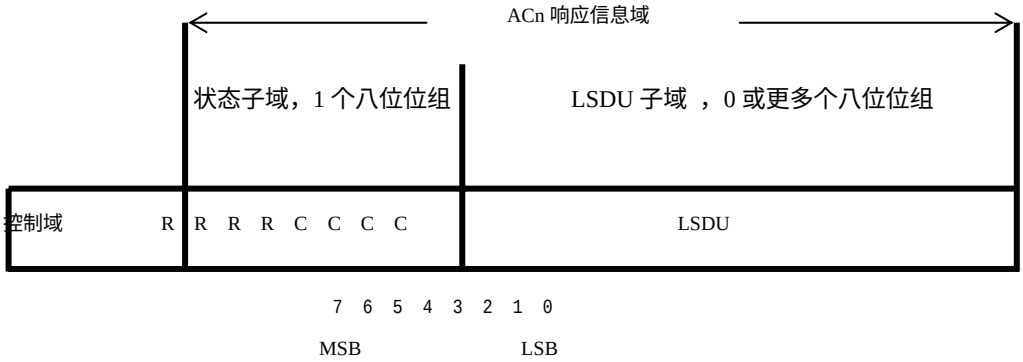


图10　ACn 响应 PDU 信息域

状态子域中 CCCC 部分的返回码，指示了命令 PDU 信息传送的成功或失败，CCCC 的可取值见表 10。

ACn 响应状态子域 CCCC 的值　　表 10

CCCC	助记符	类型	描述
0 0 0 0	OK	成功	收到命令
0 0 0 1	RS	永久错误	不可执行或未激活的服务
0 1 0 1	UE	永久错误	LLC 用户接口错误
0 1 1 0	PE	永久错误	协议错误
0 1 1 1	IP	永久错误	永久的执行依赖性错误
1 0 0 1	UN	暂时错误	源暂时不可用
1 1 1 1	T	暂时错误	暂时的执行依赖性错误
注：所有其他 CCCC 的代码值被保留。			

状态子域中 RRRR 部分返回的代码指示了响应 PDU 信息传送的成功或失败，RRRR 的可取值见表 11。

ACn 响应状态子域中 RRRR 的值　　表 11

RRRR	助记符	类型	描述
0 0 0 0	OK	成功	响应 LSDU 被提交
0 0 0 1	RS	永久错误	不可执行或未激活的服务
0 0 1 1	NE	永久错误	没有提交响应 LSDU
0 1 0 0	NR	成功	没有请求响应 LSDU
0 1 0 1	UE	永久错误	LLC 用户接口错误
0 1 1 1	IP	永久错误	永久的执行依赖性错误
1 0 0 1	UN	暂时错误	源暂时不可用
1 1 1 1	IT	暂时错误	暂时的执行依赖性错误
注：所有其他 RRRR 的代码值被保留。			

如果响应 PDU 中的 F 比特为 0，则 RRRR 子域应置成“ NR ”。

5.5.6 LLC 过程描述

5.5.6.1寻址过程

（1）类型 1 操作过程

上行链路支持专用 MAC 地址。下行链路支持专用和广播 MAC 地址。

（2）类型 3 操作过程

LID 应是专用 MAC 地址。

5.5.6.2P/F 比特的使用过程

（1）类型 1 操作过程

所有 UI 命令 PDU 发送时 P 比特置 0。

（2）类型 3 操作过程

如果一个命令 PDU 没有请求目的 LLC 在对其的确认中返回一个 LSDU，源 LLC 将该 ACn 命令 PDU 的 P 比特置 0。

如果一个命令 PDU 请求目的 LLC 在对其的确认中返回一个 LSDU，源 LLC 将该 ACn 命令 PDU 的 P 比特置 1。如果只希望数据从目的 LLC 传递给源 LLC，可将命令 PDU 的信息域置空。

当发送一个 ACn 响应 PDU 时，目的 LLC 把 F 比特置为与接收到的 ACn 命令 PDU 中的 P 比特相同的值，并且只有当 F 比特为 1 时，该 PDU 才包含一个非空的 LSDU 子域。

5.5.6.3链路建立过程

V（SI)、V（RI）和 V（RB）应当在与其相关联的链路建立（撤消）时建立（撤消)。V（SI）创建时的初值应设置为“ 0 ”。

5.5.6.4信息传送过程

（1）类型 1 操作过程

信息的传送应通过发送 P 比特设置为“ 0 ”的 UI 命令 PDU 实现。

MAC 控制域中的 C／R 比特用于标识出 PDU 中是否包含一个命令。第二层的 LLC 过程无需对接收到的 UI 命令 PDU 进行确认。

（2）类型 3 操作过程

①发送 ACn 命令

从源 LLC 到响应 LLC 的信息传输应通过发送 ACn 命令实现。发送 LLC 可以在任何时刻向任何接收 LLC 发送 ACn 命令 PDU，只要此发送 LLC 当前没有在等待来自该接收 LLC 的 ACn 响应 PDU。

当从数据链路层用户接收到一个 DL-DATA-ACK.request 时，LLC 将发送一个包含 LSDU 的 ACn 命令 PDU，其 P 比特设置为“ 0 ”。

当从数据链路层用户接收到一个 DL-REPLY.request 时，LLC 将发送一个包含 LSDU 的 ACn 命令 PDU，其 P 比特设置为“ 1 ”。

当构造 ACn 命令 PDU 时，V(SI)的值被用来确定 PDU 的 LLC 控制域代码。当 V(SI)值为“ 0 ”时，LLC 控制域代码将为 ACn，n 等于 0；当 V(SI)值为“ 1 ”

时，LLC 控制域代码将为 ACn，n 等于 1。

LLC 发送一个命令 PDU 时，将为该传送启动一个确认定时器，同时将一个内部传送计数变量加 1。如果在该确认定时器超时之前没有收到 ACn 响应 PDU，则发送 LLC 将会重新传送该命令，并将内部传送计数变量加 1，复位并重新启动确认定时器。

如果仍没有收到响应 PDU，重新发送过程将反复执行，直到内部传送计数变量的值等于逻辑链路参数 N3，此时失败状态将被报告给数据链路层用户。

②接收 ACn 命令

(a)收到 ACn 命令 PDU 时的比较

当接收到一个 ACn 命令 PDU，LLC 将对 V(RI)与接收到的 LPDU 的 LLC 控制域代码第八比特进行比较。

如果比较的结果是相等，那么认为所接收到的 PDU 是非重发 PDU；否则认为所接收到的 PDU 是最近一次接收到的 ACn 命令 PDU 的一个重发副本。

(b)非重发的 ACn 命令

如果接收到的 LPDU 有效、非空且 P 比特为 0，那么此 LSDU 将由 DL-DATA-ACK.indication 原语传递给数据链路层用户。

如果 P 比特为 1，所请求的应答 LSDU 可得，并且接收到的 LSDU 非空，则所接收到的 LSDU 将在 DL-REPLY.indication 原语中传递给数据链路层用户。

如果 P 比特为 1，所请求的应答 LSDU 不可得，并且接收到的 LSDU 非空，则所接收到的 LSDU 将在 DL-DATA-ACK.indication 原语中传递给数据链路层用户。

V(RI)应被置为接收到的 PDU 中 LLC 控制域代码第八比特的反码。

LLC 应当通过向 ACn 命令 PDU 的发起方发送一个 ACn 响应 PDU 对已收到的非复制 ACn 命令 PDU 进行确认，该 PDU 的 LLC 控制域第八比特被设置为 V(RI)的当前值。

如果接收到的命令 PDU 的 P 比特为“ 0 ”，发送的响应 PDU 应将 F 比特设置为“ 0 ”，并且其信息域中只包含状态子域。

如果接收到的命令 PDU 的 P 比特为 1，发送的响应 PDU 应将 F 比特设置为 1；并且如果该 LSDU 可得，则在该 PDU 信息域中应包含此 LSDU。

(c)重发的 ACn 命令

除了以下例外情况，收到重发 ACn 命令 PDU 时的 LLC 过程与接收到非重发 PDU 的 LLC 过程相同。

- 收到一个重发的命令 PDU 将不会对 V(RI)和 V(RB)状态变量产生影响；
- 不管命令 PDU 中的 P 比特为何值，都不会发送 DL-DATA-ACK.indication 原语；
- 如果在命令 PDU 中收到了一个 LSDU，将被丢弃。

③发送 ACn 响应

只有收到一个 n 等于 1 的 ACn 命令的时侯，n 等于 0 的 ACn 响应 PDU 才会被发出。

只有收到一个 n 等于 0 的 ACn 命令的时侯，n 等于 1 的 ACn 响应 PDU 才会被发出。

该响应应被发送到与之相关的命令 PDU 的发送端。

响应 PDU 中的状态子域应指示资源是否空闲可用，使其成功的接收了与之

关联的命令 PDU 中的信息域，以及在 F 比特为“ 1 ”的情况下，所请求的 LSDU 是否就绪，以在响应 PDU 中被返回。

ACn 响应 PDU 中状态子域 CCCC 部分状态码的设置应参照此前存储在相应 V(RB)状态变量中的接收状态。

④接收确认

当传送一个 ACn 命令 PDU 给某个目的 LLC 后，源 LLC 将期待从该目的 LLC 处收到一个 ACn PDU 形式的确认。

n 等于 0 的 ACn 命令应接收到 n 等于 1 的 ACn 确认，反之亦然。

接收到一个响应 PDU 后，LLC 将对响应 PDU 中 LLC 控制域代码的第八比特和传送序列状态变量 V(SI)的当前值进行比较。

如果比较结果为不等，则认为该响应是有效的，LLC 将停止与之关联的确认定时器，将内部传送计数器复位至“ 0 ”。V(SI)状态变量将被取反。

LLC 将发送一个 DL-DATA-ACK-STATUS.indication 原语或者 DL-REPLY-STATUS.indication 原语给数据链路层用户，发送何种原语取决于当前是何种请求原语正在等待确认。当响应数据在 ACn 响应 PDU 中被返回时，包含响应数据的 LSDU 将被传递给数据链路层用户。

LLC 应根据响应 PDU 中状态子域的内容，将状态信息传递给数据链路层用户。

如果响应 PDU 中 LLC 控制域代码的第八比特和传送序列状态变量 V(SI)的当前值的比较结果为相等，则认为该响应 PDU 是无效的。LLC 将不会执行进一步的操作，同时继续等待收到一个有效的 ACn 响应 PDU。确认定时器不会受到任何影响。

5.5.6.5逻辑链路参数

（1）PDU 中最大八位位组数 N2

N2 是一个逻辑链路参数，它表示一个 PDU 中的八位位组的最大数目。

（2）PDU 中最小八位位组数

最小长度的有效命令 PDU 应包含控制域。因此，一个有效命令 PDU 的最小八位位组数应为 1。

最小长度的有效 ACn 响应 PDU 应按序包含控制域和状态子域。因此，一个有效响应 PDU 的最小八位位组数应为 2。

（3）信息发送最大次数 N3

N3 是一个逻辑链路参数，它指出了 LLC 为了完成一次信息交换所进行的 ACn 命令 PDU 发送的最大次数。通常，N11 被设置为足够大以克服由于链路错误所造成的 PDU 的丢失。N3 的值也可能设置为 1，这样 LLC 子层就不会将一个 PDU 重新交给 MAC 子层。

（4）确认时间 N4

确认时间决定确认定时器的周期，并由此定义源 LLC 期望从目的 LLC 接收到 ACn 响应 PDU 的最大等待时间。确认时间应考虑到 MAC 子层引入的延时，以及定时器的启动是在命令 PDU 传送开始时还是在命令 PDU 传送结束时。正确的操作过程要求此确认时间大于 ACn 命令 PDU 发送与相关的 ACn 响应 PDU 接收之间的正常时间间隔。RSU 和 OBU 中的 N4 取值可能不同。

6 应用层

6.1 应用层核心架构

应用层核心包含T-KE、B-KE、I-KE，T-KE提供I-KE以及应用所需的数据传输基础。应用层核心架构见图11。

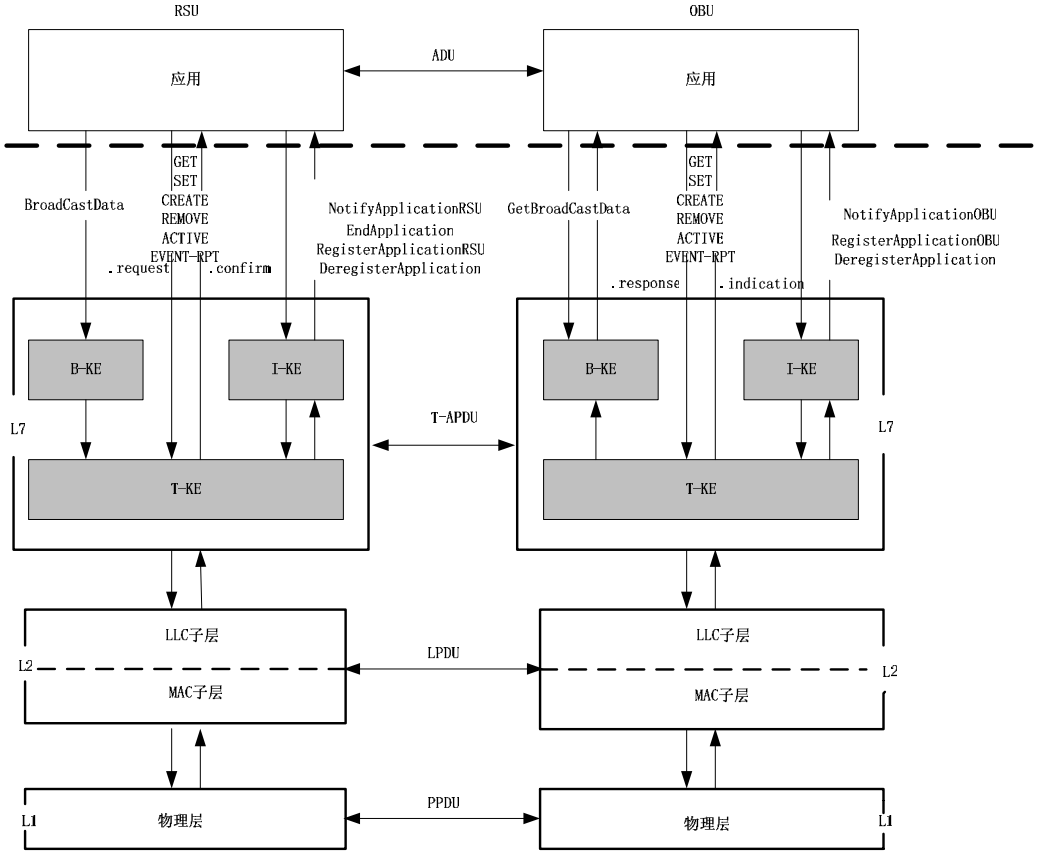


图11 应用层核心架构

6.2 T-KE

6.2.1 功能

T-KE 通过将预定义的服务原语转换成 T-APDU 及其逆过程，在两个服务用户之间传送信息，是对传送的具体实现的抽象表示。

6.2.2 服务

6.2.2.1范围

T-KE 应提供表 12 所列的服务。

T-KE 服务 **表 12**

序号	服务	服务描述
1	GET	由应用启用，用以读出对方应用信息数据文件。只能在确认模式下才能请求该

序号	服务	服务描述
		服务，并要求应答
2	SET	由应用启用，用以更新对方应用信息数据文件。可在确认模式或不确认模式下请求该服务。在确认模式下要求应答
3	CREATE	由应用启用，用以创建对方应用的数据格式化信息（目录与文件）。只能在确认模式下才能请求该服务，并要求应答
4	REMOVE	由应用启用，用以删除对方应用的数据格式化信息（目录与文件）。只能在确认模式下才能请求该服务，并要求应答
5	ACTION	由应用启用，要求对方应用完成某特定操作。操作由操作类型值进一步限定。可在确认模式或不确认模式下请求该服务，在确认模式下要求应答
6	EVENT-REPORT	由应用或I-KE启用，实现向同级应用或I-KE报告事件。在确认模式下要求应答
7	INITIALIZATION	由I-KE启用，对RSU和未与其建立通信的每个OBU之间的通信进行初始化。初始化服务只应由I-KE使用

6.2.2.2服务原语

T-KE 应由下列服务原语提供服务：

a) GET.request，GET.indication，GET.response，GET.confirm；
b) SET.request，SET.indication，SET.response，SET.confirm；
c) CREATE.request，CREATE.indication，CREATE.response，CREATE.confirm；
d) REMOVE.request，REMOVE.indication， REMOVE.response，REMOVE.confirm；
e) ACTION.request，ACTION.indication，ACTION.response，ACTION.confirm；
f) EVENT-REPORT.request，EVENT-REPORT.indication，EVENT-REPORT.response，EVENT-REPORT.confirm；
g) INITIALIZATION.request，INITIALIZATION.indication，INITIALIZATION.response，INITIALIZATION.confirm。

其中：INITIALIZATION.request 和 INITIALIZATION.confirm 原语应在 RSU 端使用，INITIALIZATION.indication 和 INITIALIZATION.response 原语应在 OBU 端使用。确认模式和不确认模式下使用的服务见图 12 和图 13。

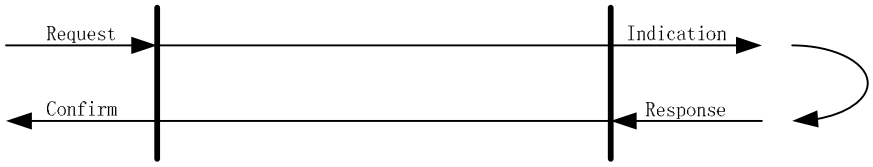


图12　确认模式下使用的服务

图13　不确认模式下使用的服务

6.2.2.3服务原语格式

服务原语的 T-ASDU 应具有表 13~表 19 的格式。

GET 原语 **表 13**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型[c]
启用标识	IID	可选	可选	=[a]	=	Dsrc-DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID [b]	IID／DID	Dsrc-DID
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
文件标识	FID	必备	必备	必备	必备	FID
偏移量	Offset	必备	必备	—	—	INTEGER
长度	Length	必备	必备	—	—	INTEGER
数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
文件	FileContent	—	—	可选	可选	File
返回码	Ret	—	—	必备	必备	ReturnStatus
注：对记录型文件，Offset表示记录标识号，Length表示记录内容长度。						
a 与相应的请求/指示相同，下同。 b 必备。若出现IID 则予以相关指示，否则予以DID的相关指示，下同。 c ASN.1类型见附录A，下同。						

SET 原语 **表 14**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型
启用标识	IID	可选	可选	=	=	Dsrc-DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID	IID／DID	DID
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
文件标识	FID	必备	必备	必备	必备	FID
偏移量[a]	Offset	必备	必备	—	—	INTEGER
长度	Length	必备	必备	—	—	INTEGER
文件	FileContent	必备	必备	—	—	File
模式	Mode	必备	必备	—	—	BOOLEAN
数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
返回码	Ret	—	—	必备	必备	ReturnStatus
a 对记录性文件的写入，都是增加记录形式；如果检测到与上一条记录内容相同，则不做操作。						

CREATE 原语 **表 15**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型
启用标识	IID	可选	可选	=	=	Dsrc-DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID	IID／DID	Dsrc-DID
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
文件清单	FileList	必备	必备	—	—	FileList

数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
返回码	Ret	—	—	必备	必备	ReturnStatus
注：支持两种文件的创建：二进制文件和记录型文件（可变长记录）。						

REMOVE 原语 **表 16**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型
启用标识	IID	可选	可选	=	=	Dsrc-DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID	IID／DID	Dsrc-DID
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
文件标识清单	FileIdList	必备	必备	—	—	FileIdList
数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
返回码	Ret	—	—	必备	必备	ReturnStatus

ACTION 原语 **表 17**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型
启用标识	IID	可选	可选	=	=	DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID	IID／DID	DID
操作类型	ActionType	必备	必备	—	—	INTEGER(0..127,...)
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
操作参数	ActionParameter	可选	可选	—	—	Container
模式	Mode	必备	必备	—	—	BOOLEAN
数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
响应参数	ResponseParameter	—	—	可选	可选	Container
返回码	Ret	—	—	可选	可选	ReturnStatus

EVENT -REPORT 原语 **表 18**

参数名称	英文表示	请求	指示	响应	确认	ASN.1 类型
启用标识	IID	可选	可选	=	=	DID
链路标识	LID	必备	必备	=	=	INTEGER
链接	Chaining	必备	必备	=	=	BOOLEAN
目录标识	DID	必备	必备	IID／DID	IID／DID	DID
事件类型	EventType	必备	必备	—	—	INTEGER(0..127,...)
访问凭证	AccessCredentials	可选	可选	—	—	OCTET STRING
事件参数	EventParameter	可选	可选	—	—	Container
模式	Mode	必备	必备	—	—	BOOLEAN
数据流控制	FlowControl	必备	必备	必备	可选	INTEGER
返回码	Ret	—	—	可选	可选	ReturnStatus

INITIALIZATION 原语 **表 19**

参数名称	英文表示	请求	指示	响应	确认	ASN.1类型
链路标识	LID	必备	必备	必备	必备	INTEGER
初始化参数	InitializationParameter	必备	必备	必备	必备	BST/VST

6.2.2.4参数的设定和释义

（1）启用标识

服务启用者所对应的 ASN.1 型专用短程通信目录标识。若响应送达缺省的启用者，则不需要该参数；若使用 IID，应包含响应此原语的 DID。

（2）链路标识

OBU 的 I-KE 所选择的 LID；LID 取值与 MAC 地址相同。

（3）链接（Chaining）

布尔型参数，若取值为“ TRUE ”，则执行 6.2.3.9。

（4）目录标识

接收方的 ASN.1 型专用短程通信目录标识，接收方的 T-KE 用 DID 向所述目录提交指示或确认。

（5）访问凭证

ASN.1 型八位位组字符串，带有满足访问条件所需的安全性相关信息，用以在指定目录上进行操作。

（6）文件标识

接收 GET.indication 的目录的文件标识。访问条件得到满足，文件内容通过 GET.response 和 GET.confirm 送达启用 GET.request 的目录。

（7）偏移量（Offset）

在 GET 服务中是数据内容在二进制文件中的起始位置或者记录型文件的 RID。

在 SET 服务中是数据内容在二进制文件中的起始位置或者记录型文件的 RID。

（8）数据流控制（FlowControl）

表示基础通信服务行为的参数，由 T-KE 映射到某个 LLC 服务上，LLC 服务见第五章。数据流控制参数、行为和 LLC 服务之间的关系见表 20。

数据流控制参数、行为和 LLC 服务间的关系　　表 20

数据流控制	行为	LLC服务
1	无流控制，无应答	不带响应请求的DL-UNITDATA.request
2	无流控制，有应答	带响应请求的DL-UNITDATA.request
3	无流控制	DL-UNITDATA.indication
4	流控制，数据单元传输	DL-DATA-ACK.request
5	流控制，数据单元传输	DL-DATA-ACK.indication
6	流控制，数据单元传输状态	DL-DATA-ACK-STATUS.indication
7	流控制，数据单元交换	DL-REPLY.request
8	流控制，数据单元交换	DL-REPLY.indication
9	流控制，数据单元交换状态	DL-REPLY-STATUS.indication
10	流控制，数据单元交换准备	DL-REPLY-UPDATE.request
11	流控制，数据单元交换准备状态	DL-REPLY-UPDATE-STATUS.indication

（9）文件、文件清单和文件标识清单

文件是SET.request／SET.indication或GET.response ／GET.confirm发送的文件内容。若访问条件满足，接收SET.indication的目录应将文件标识中识别的文件内容修改为文件中给定的内容值。在GET.response/GET.confirm的情况下，若访问条件得到满足，收到相应GET.indication的目录应将GET.indication

的文件标识文件内容值发送给启用GET.request的目录。

文件清单是包含文件标识和文件长度信息的列表，文件标识清单是文件标识信息的列表。

（10）返回码

对服务原语的指示的回答发出的返回代码。预定义的代码如下：

① NoErro：请求的操作执行成功；

② AccessDenied：请求的操作由于系统安全性的原因未执行；

③ ArgumentError：文件内容访问失败，原因是未认出规定文件内容，或规定文件内容超出了范

围或对文件某些内容不适合，或启用的事件报告不被接收实体支持；

④ ComplexityLimitation：请求的操作由于参数太复杂而未执行；

⑤ ProcessingFailure：操作处理遇到的一般性失败；

⑥ Processing：请求的操作正在处理，但结果不能用；

⑦ ChainingError：请求的操作按 6.2.3.9 中定义的规则未执行。

（11）模式

布尔型参数。若取值为“ TRUE ”，则服务原语的指示有服务原语的响应。

（12）操作类型

用以标识针对接收方目录的特定操作。

（13）操作参数

启用 ACTION 操作所需的信息。

（14）响应参数

执行 ACTION 操作而产生的结果信息。

（15）事件类型

向接收 EVENT-REPORT.indication 的目录提交的消息。

（16）事件参数

分别通过 EVENT-REPORT.request 和 EVENT-REPORT.indication 发送消息所需的附加信息。

（17）初始化参数

通过初始化服务发送，通信初始化所需的信息。

6.2.3 协议规程

6.2.3.1步骤

T-KE 的传送过程由以下步骤组成，其运行顺序见图 14。

（1） 将 SDU 转换为 PDU；

（2） 将 PDU 编码；

（3） 八位位组对齐；

（4） 多路复用、拼接和 LLC 访问；

（5） 解多路复用；

（6） PDU 解码、解拼接和去除插入的“ 0 ”位；

（7） PDU 转换为 SDU，并按收件人分发。

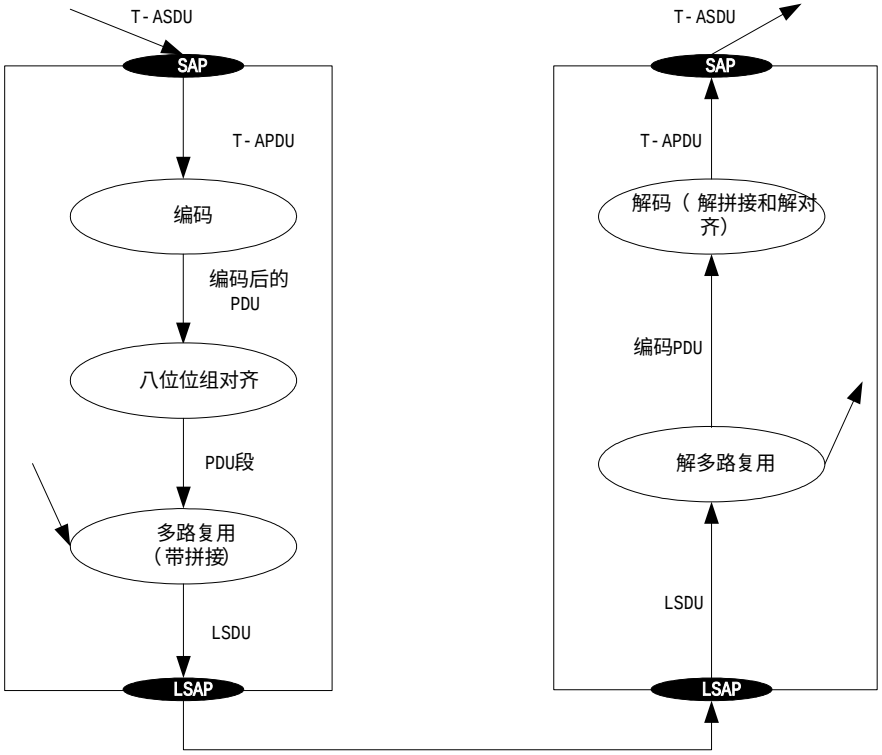


图14　T-KE 协议

6.2.3.2 SDU 转换 PDU

T-KE 根据下列规则将请求和响应服务原语转换为 T-APDU：

（1）　服务请求转换为附录 A 中规定的相应的服务请求 T-APDU；

（2）　服务响应转换为附录 A 中规定的相应的服务响应 T-APDU；

（3）　在 T-APDU 中，LID 应被去除，应通过 6.3.7 中规定的每个 LLC 服务原语转交给 LLC。在 INITIALIZATION.request 的情况下，LID 的值应为 0xFFFFFFFF。

将 SDU 转换为 PDU 的过程见图 15。

```
SET.request(IID,LID,Chaining,DID,AccessCredentials,FID,Offset,Length,Mode,FlowControl)
                                                    ↓
Set-Request::=SEQUENCE{
          fill                BIT STRING (SIZE(1)),
          mode                BOOLEAN,
          did                 Dsrc-DID,.
          accessCredentials   OCTET STRING (SIZE(0..127,...)) OPTIONAL,
          fileid              FileID,
          offset              Integer(0..32767,...),
          length              Integer(0..127,...),
          iid                 Dsrc-DID OPTIONAL
          }
```

图15　SDU 转换为 PDU

6.2.3.3 编码

T-KE 应按照 GB/T16263.2 的规定对请求和响应 PDU 进行编码。可编码的

ASN.1 表示法见 6.5。

编码过程见图 16。

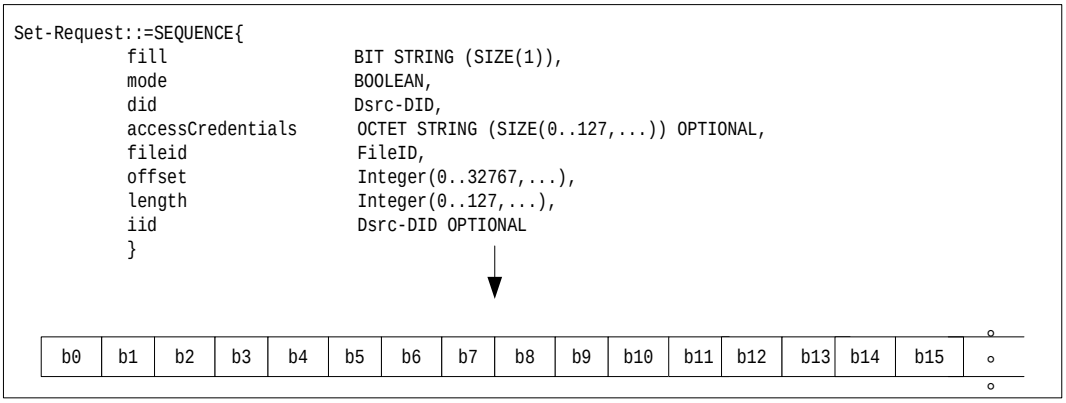


图16　编码

6.2.3.4分段字头

分段字头应包含分段指示符、PDU 编号、分段计数器和分段编号扩展指示符组成。相关比特的位置见图 17。各比特按从 7 到 0 的顺序编号，其中 7 是最高有效位，0 是最低有效位。

7	6	5	4	3	2	1	0
分段指示符	PDU编号				分段计数器		扩展指示符

图17　分段字头

分段指示符应为1，表示该PDU未进行分段。PDU编号0000和0001应只由B-KE发送的T-APDU分段所使用。扩展指示符应为1。

6.2.3.5八位组对齐

八位位组对齐有两种机制：

（1）　通过在 T-APDU 的 ASN.1 定义中插入填充比特来对齐，见附录 A。建议使用“ 0 ”作为填充比特；

（2）　T-KE 对分段进行补“ 0 ”操作，直到总的比特数达到 8 的倍数。所插入的“ 0 ”的数目应在 0～7 之间。

八位位组对齐过程见图 18。

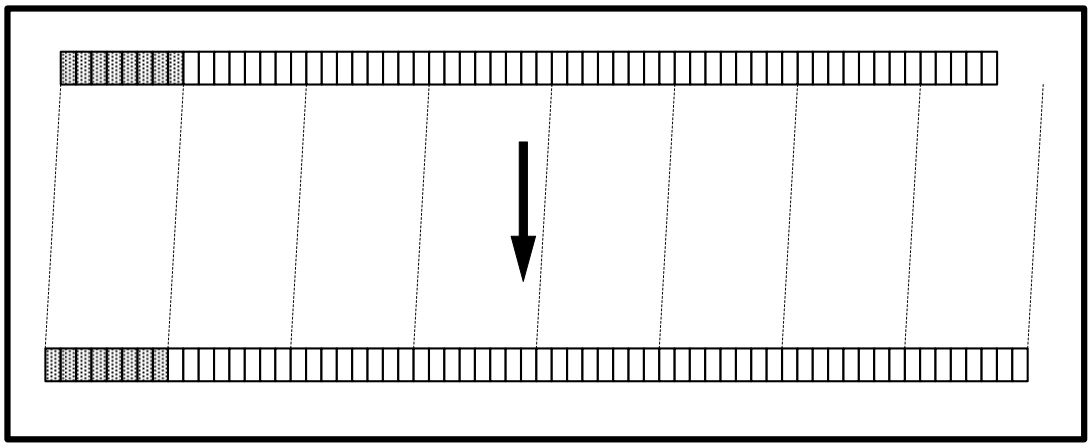


图18　八位位组对齐

6.2.3.6多路复用

T-KE 应按照固定优先权队列对 T-APDU 分段进行复用，优先权由 I-KE 给出（见 6.3.2.2 和 6.3.2.3）。

T-ASDU 分段的复用过程见图 19。

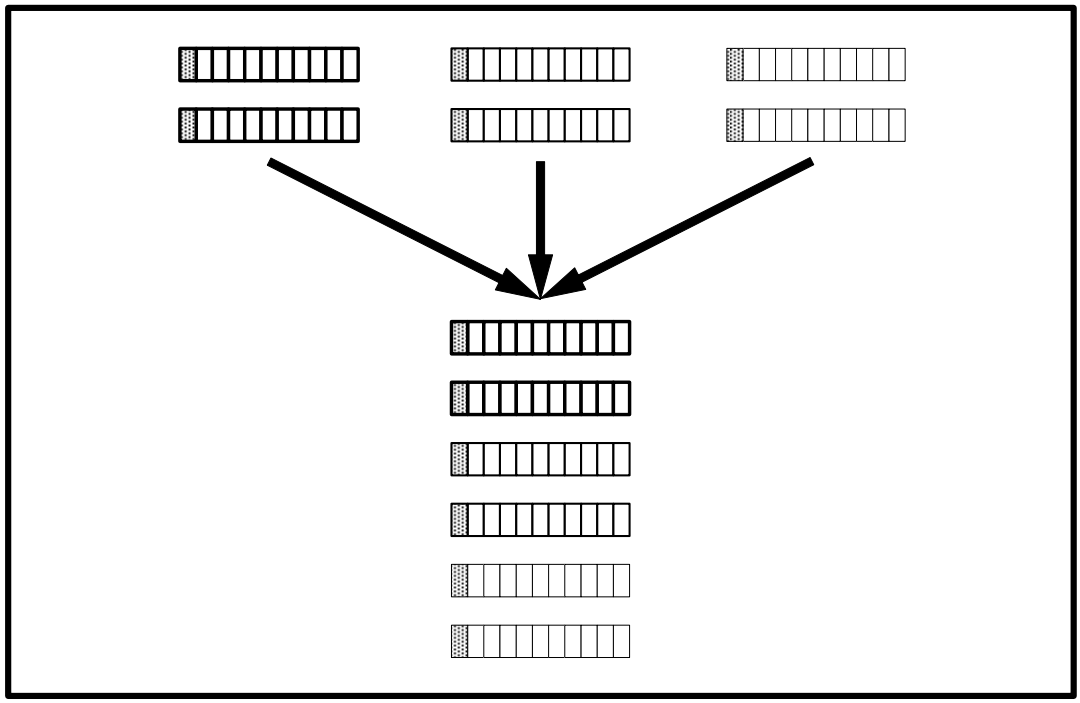


图19　多路复用

6.2.3.7访问 LLC

T-KE 应使用 T-APDU 的数据流控制参数中所指定的 LLC 服务。数据流控制参数应按 6.2.2.4 的（8）中的规定进行解释。LLC 的访问过程见图 20。

对 INITIALIZATION.request 服务，应使用带有响应请求的 DL-UNITDATA.Request 服务，对于 INITIALIZATION.response 服务，应使用不带响应请求的 DL-UNITDATA.Request 服务。

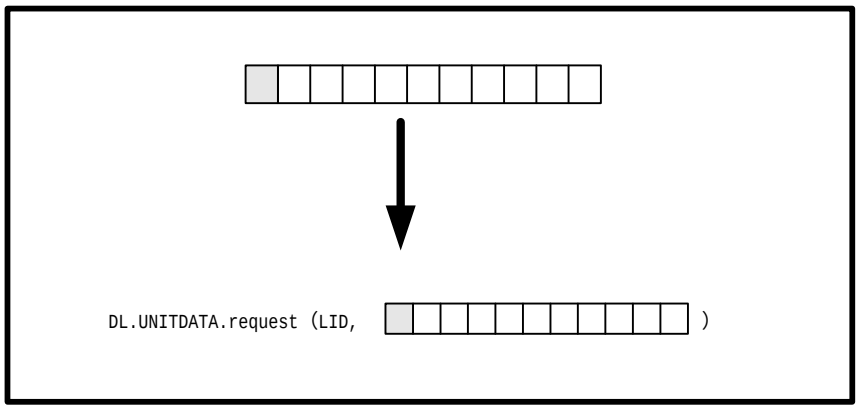


图20　LLC 的访问

6.2.3.8拼接

如果使用 LLC 服务及其 LID 都相同，且不超过 LLC 最大帧的长度，多个连续 T-APDU 可被映射到一个 LLC 服务上进行拼接。在 LSDU 中，T-APDU 分段的顺序应当由 6.3.6 给定。拼接过程见图 21。

拼接的情况仅在一个或多个短的未分段的 T-APDU 被映射到一个 LLC 服务上时出现。

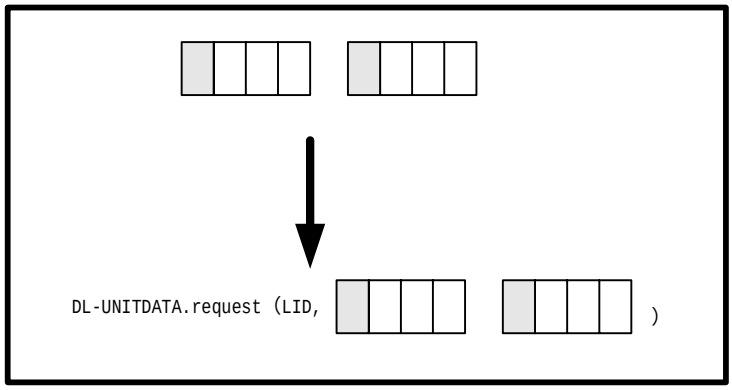


图21　拼接

6.2.3.9带有链接的拼接

有相同 PDU 编号的连续有序拼接的 T-APDU 组成一个链接。

同一个链接的T-APDU所调用的操作的执行应取决于链中先前的T-APDU所调用的操作是否已成功执行。

如果一个已被链接的 T-APDU 生成的响应带有非“ 无错误 ”返回状态，则同一链中后续 T-APDU 所调用的所有操作都不应执行，并且相关的响应均应包含“ 链接错误 ”的返回状态。

6.2.3.10　解复用

T-KE 应根据包含在分段字头中的 PDU 编号，将 LLC 指示原语的数据域中收到的 T-APDU 分段进行解复用。已拼接的分段应按第一个分段字头中的 PDU 编号进行解复用。

解复用的详细过程见图 22。

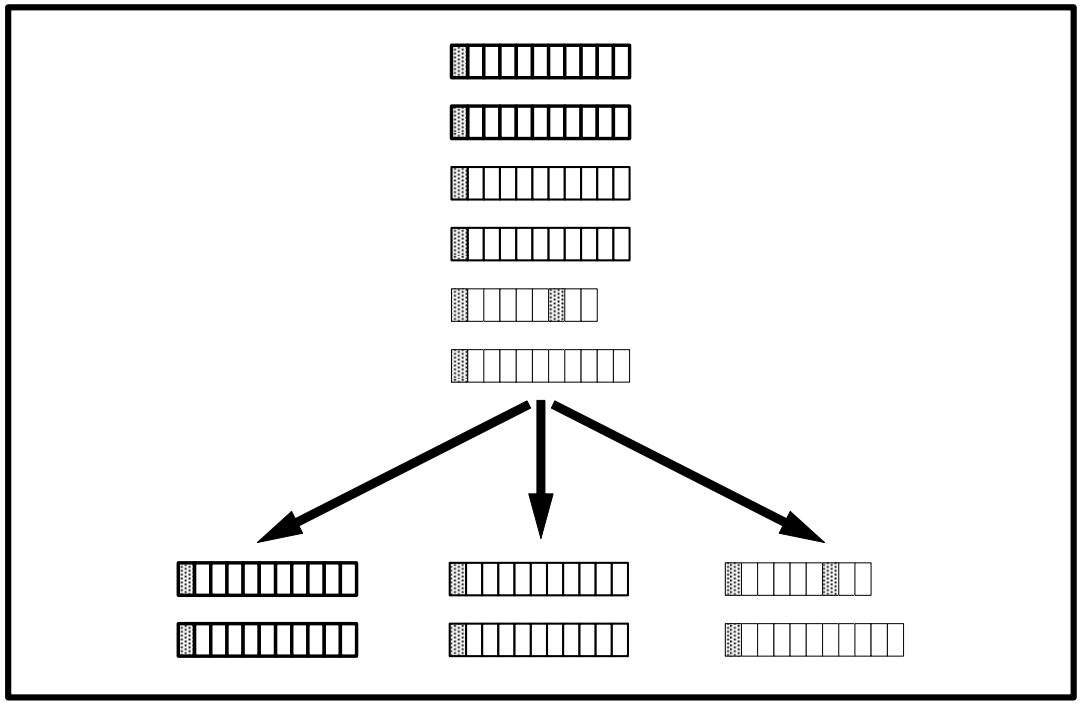


图22　解复用

6.2.3.11　解码

T-KE 应按照 GB/T16263.2 中规定的 ASN.1 编码规则 第 2 部分 :紧缩编码规则（PER）规范的规定，将已并段的 T-APDU 进行解码。可解码的 ASN.1 类型见附录 A。

1 至 7 比特的拖尾“ 0 ”比特的接收不应引起差错。

如果接收到了多于七个拖尾比特，且顺序排列值为“ 0 ”的比特的个数不多于 7，则应将这些“ 0 ”比特去除（因为这些比特是为了实现八位位组的对齐而被插入的）。如果后续字节第 3 至 1 比特的值为 001， 则该字节应去除。据此规定，剩下的比特应作为 T-APDU 予以解码。

在所有其他情况下，已并段的 T-APDU 都应被抛弃。

如果 T-KE 不能对已并段的 T-APDU 解码，则已并段的 T-APDU 应被抛弃。T-KE 不得对附录 A 中在容器定义中所定义的虚拟 T-APDU 予以解码。

解码的详细过程见图 23。

b0	b1	b2	b3	b4	b5	b6	b7	b8	b9	b10	b11	b12	b13	b14	b15

```
Set-Request::=SEQUENCE{
            fill                 BIT STRING (SIZE(1)),
            mode                 BOOLEAN,
            did                  Dsrc-DID,
            accessCredentials    OCTET STRING (SIZE(0..127,...)) OPTIONAL,
            fileid               FileID,
            offset               Integer(0..32767,..),
            length               Integer(0..127,..),
            iid                  Dsrc-DID OPTIONAL
            }
```

图23　解码

6.2.3.12　PDU 转换为 SDU

已解码的 T-APDU 应当根据下列规则来构建 T-ASDU：

（1） 服务请求应当转换为相应的服务指示 T-ASDU；

（2） 服务响应应当被转换为相应的服务确认 T-ASDU；

（3） T-ASDU 应当递交给 T-APDU 的 Dsrc-DID 参数中所寻址到的目录。INITIALIZATION.indication 应递交给 I-KE；

（4） 如果所寻址的目录不存在，则 T-ASDU 应当被抛弃；

（5） T-KE 应将此 SDU 的 LID 通知给层管理。

PDU 转换为 SDU 的过程见图 24。

```
Set-Request::=SEQUENCE{
          fill                  BIT STRING (SIZE(1)),
          mode                  BOOLEAN,
          dirID                 Dsrc-DID,
          accessCredentials     OCTET STRING (SIZE(0..127,...)) OPTIONAL,
          fileid                FileID,
          offset                INTEGER(0..32767,...),
          length                INTEGER(0..127,..),
          iid                   Dsrc-DID OPTIONAL
          }
                                    ↓
SET.request(IID,LID,Chaining,DID,AccessCredentials,FID,Offset,Length,Mode,FlowControl)
```

图24　PDU 转换为 SDU

6.3　I-KE

6.3.1 总则

I-KE 应：

（1） 通过与同级实体间有关配置或应用的信息交换，实现 OBU 和 RSU 之间通信初始化；

（2） 按照 6.3.2 规定的服务原语提供服务；

（3） 按照 6.5 规定的 BST 进行通信的初始化，并将 VST 传送到一个 LLC 服务原语中；

（4） 按照 6.5 规定的 VST 进行通信的初始化；

（5） 按照 6.3.3 规定的协议规程和行为进行初始化。

6.3.2 服务

6.3.2.1服务原语

I-KE 应提供的服务表 21。

I-KE 服务　　**表 21**

序号	服务原语	功能说明
1	RegisterApplicationRSU	RSU 端的应用启用服务，实现在 I-KE 应用清单中注册此应用者

2	RegisterApplicationOBU	OBU 端的应用启用的服务，实现在 I-KE 应用清单中注册此应用
3	DeregisterApplication	应用启用的服务，实现对应用清单中相关注册项的删除
4	NotifyApplicationOBU	I-KE 用于通知 OBU 端的应用关于可能的通信参与者的出现情况和 OBU 生成 LID 的情况
5	NotifyApplicationRSU	I-KE 用于通知 RSU 端的应用关于可能的通信参与者和关联 OBU 的 LID 的情况
6	EndApplication	应用启用 EndApplication 服务来通知 I-KE 该应用不再需要 LID

6.3.2.2服务原语的格式

服务原语的格式见表 22～表 27。

RegisterApplicationRSU 原语 **表 22**

参数名称	英文表示	ASN.1 型	参数说明
应用标识	AID	DSRCAppllicationEntityID	—
必备应用	MandatoryApplication	BOOLEAN	—
优先级	Priority	INTEGER	—
目录标识	DID	Dsrc-DID	可选
配置	Profiles	SEQUENCE OF Profile	可选
应用参数	ApplicationParameter	ApplicationContextMark	可选

RegisterApplicationOBU 原语 **表 23**

参数名称	英文表示	ASN.1 型	参数说明
应用标识	AID	DSRCAppllicationEntityID	—
必备应用	MandatoryApplication	BOOLEAN	—
优先级	Priority	INTEGER	—
目录标识	DID	Dsrc-DID	可选
配置	Profiles	SEQUENCE OF Profile	可选
应用参数	ApplicationParameter	ApplicationContextMark	可选

DeregisterApplication 原语 **表 24**

参数名称	英文表示	ASN.1 型	参数说明
应用标识	AID	DSRCAppllicationEntityID	—
目录标识	DID	Dsrc-DID	可选

NotifyApplicationRSU 原语 **表 25**

参数名称	英文表示	ASN.1 型	参数说明
优先级	Priority	INTEGER	—
目录标识	DID	Dsrc-DID	若 AID 在 VST 中出现
链路标识	LID	INTEGER	—
应用参数	ApplicationParameter	ApplicationContextMark	可选
OBU 配置	obuConfiguration	ObuConfiguration	—

NotifyApplicationOBU 原语 **表 26**

参数名称	英文表示	ASN.1 型	参数说明
路侧单元标识	RSUID	BeaconID	—
优先级	Priority	INTEGER	—

目录标识	DID	Dsrc-DID	若 AID 在 BST 中出现
链路标识	LID	INTEGER	—
应用参数	ApplicationParameter	ApplicationContextMark	可选

EndApplication 原语 **表 27**

参数名称	英文表示	ASN.1 型	参数说明
目录标识	DID	Dsrc-DID	—
链路标识	LID	INTEGER	—

6.3.2.3参数的设定与释义

（1）应用标识

用以识别 DSRC 的应用。

（2）必备应用

若应用是必备的，则为“ TRUE ”，若是可选的，则为“ FALSE ”。

（3）优先级

应用的优先级别。数值越小优先级越高。

（4）目录标识

目录标识在：

①RegisterApplicationRSU 服务中,识别将注册的目录。若为单一缺省应用，应不使用 DID；

②RegisterApplicationOBU 服务中，识别将注册的目录；

③NotifyApplication 服务中，将识别同一级别的应用。

（5）配置

若有，应列出与应用相关联的配置清单。

（6）应用参数

若有，应分别在 NotifyApplicationOBU 或 NotifyApplicationRSU 服务中传送。

（7）路侧单元标识

提供服务的 RSU 的识别标识。

（8）OBU 配置

指出 NotifyApplicationRSU 中给定的与 LID 相关的 OBU 的配置和状态。

6.3.2.4命名和登记

DSRC 应用层的 ManufacturerID 和 MAC 地址应以唯一的标识来登记。

应用标准应保证下列数据项的统一性。

——ActionType：应是操作唯一的标识；

——Dsrc-DID：应是在每个应用中是唯一的；

——EventType：应是事件唯一的通用标识。

6.3.3 协议规程

6.3.3.1RSU：BST 的重复传输

在 RSU 端，I-KE 应传送附录 A 定义的 BST。应使用 T-KE 带有初始化参数为 BST 的 INITIALIZATION.request 服务。

6.3.3.2OBU：BST 的接收和 VST 的传输

在 OBU 端，I-KE 通过 T-KE 的带有初始化参数为 BST 的 INITIALZATION.indication 来接收 BST。

只要下列条件之一得到满足：

(1) BeaconID 与最近一次收到的 BeaconID 不同；

(2) 上一次收到的 BeaconID 与目前收到的 BeaconID 之间的时间超过了 255 秒。

则 OBU 的 I-KE 应包含下列操作：

(1) I-KE 应将 BST 中收到的配置告知层管理，则 DSRC 通信以此为缺省配置；

(2) I-KE 应将 BST 内 ApplicationList 中给定的 DSRCApplicationEntityIDs 与注册的 DSRCApplicationEntityIDs 比较，如果 BST 内的 DSRCApplicationEntityIDs 是注册的，I-KE 应：

①将 DSRCApplicationEntityIDs 以及 DID 和应用参数（如果在相关的 RegisterApplicationOBU 中出现）加入到 VST 的 ApplicationList 中；

②通过 NotifyApplicationOBU 将下列信息通知应用层用户：

(a)参数 RSUID 是 BST 中给定的 BeaconID；

(b) 参数 Priority 是在 BST 必备 Application List 中 DSRCApplicationEntityID 的位置；对可选 Application List 中所有的 DSRCApplicationEntityIDs 来说，是必备 Application List 中 DSRCApplicationEntityIDs 的个数加上注册的优先级；

(c)参数 LID 是 OBU 的 MAC 地址；

(d) 参数 ApplicationParameter 若出现，则是 BST 中收到的 ApplicationParameter，否则空缺。

③I-KE 应通过层管理将已通知的应用优先级通知给 T-KE；

④I-KE 应从 BST 的 Profile 或 ProfileList 中选择 OBU 所支持的配置，并应当将 VST 中的 Profile 设置为此配置；

⑤VST 见附录 A 所示的 ASN.1 定义；

⑥I-KE 使用 T-KE 中带有下列设定参数的 INITIALIZATION.response 服务来传送 VST：

- LID；
- VST。

⑦I-KE 应保存 BeaconID 和时间；

⑧I-KE 应保存 VST 的 ApplicationList 和 LID。

6.3.3.3RSU：对 VST 的应答

RSU 的 I-KE 应用 T-KE 中带有下列设定参数的 INITIALIZATION.confirm 来

接收 VST：

（1）LID；

（2）VST。

I-KE 应使用带有下列参数的 NotifyApplicationRSU 来通知应用层用户：

（1） Priority，对于必备应用中的 DSRCApplicationEntityIDs，是必备应用清单中的位置；对可选应用中 DSRCApplicationEntityIDs，是必备应用中 DSRCApplicationEntityIDs 的个数加上在 nonmandApplications 中的位置；

（2） DID，若出现，则是 VST 中的 DID，否则空缺；

（3） LID，INITIALIZATION.confirm 中收到的 LID；

（4） ApplicationParameter，若出现，则是 VST 中的应用参数，否则空缺；

（5） obuConfiguration，VST 中的 ObuConfiguration。

I-KE 应将 LID 和 VST 中给定的配置之间的关系通知给层管理，且在 OBU 与该 LID 的后续通信中使用。

I-KE 应存储 VST 的 ApplicationList 和 INITIALIZATION.confirm 中给定的 LID。

6.3.3.4RSU：RegisterApplicationRSU

接收到 RegisterApplicationRSU 原语，RSU 的 I-KE 应向必备或可选应用中分别插入原语中给定的信息。它使用参数“必备应用”和“优先级”中给定的信息。应向 BST 的 ProfileList 中插入配置。

6.3.3.5OBU：RegisterApplicationOBU

接收到 RegisterApplicationOBU 原语，OBU 的 I-KE 应将此应用加到已注册的应用清单中。

OBU 的每个目录应具有唯一的 DID（DID=0 已保留）。

6.3.3.6OBU：DeregisterApplication

接收到 DeregisterApplication，I-KE 应将此应用从已注册的应用清单中删除。

6.3.3.7RSU：DeregisterApplication

接收到 DeregisterApplication,I-KE 应将注册项从 BST 的应用清单中删除。

6.3.3.8RSU：释放应用

接收到 EndApplication，I-KE 应将从 VST 中获得的应用清单移除。若清单为空（如所有应用被删除），I-KE 应向同级 I-KE 传送释放。它应使用带有下列参数的 T-KE 的 EVENT-REPORT.request：

（1） IID（空缺）；

(2) LID；
(3) DID 等于 0；
(4) EventType 等于 Release（0）；
(5) EventParameter（空缺）；
(6) Mode 等于 FALSE；
(7) FlowControl 等于 1。

6.3.3.9OBU：释放的接收

I-KE 应以带有下列参数的 EVENT-REPORT.indication 来接收释放：
(1) IID（空缺）；
(2) LID；
(3) DID 等于 0；
(4) EventType 等于 Release（0）；
(5) EventParameter（空缺）；
(6) Mode 等于 FALSE。
I-KE 应删除 VST。

6.4 B-KE

6.4.1 总则

B-KE应：
(1) 通过 OBU 和 RSU 的广播群组交换，为 RSU 和 OBU 的各种应用实现广播信息分发、收集；
(2) 按照 6.4.2 规定的服务原语提供服务；
(3) 发送 6.5 中规定广播群组实现通信；
(4) 按照 6.4.3 规定的协议规程实现广播。

6.4.2 服务

6.4.2.1服务原语

B-KE 应由下列服务原语提供服务：

(1) BroadcastData：只由 RSU 端的应用启用，实现向 OBU 端的同级应用文件以记录形式广播信息；
(2) GetBroadcastData：只由 OBU 端的应用启用，实现对广播数据的获取，包括 GetBroadcastData.resquest 和 GetBroadcastData.confirm。

6.4.2.2服务原语的格式

服务原语的格式见表 28 和表 29。

BroadcastData 原语 **表 28**

参数名称	英文表示	可选/必备	ASN.1 型
目录标识	DID	必备	Dsrc-DID

文件	File	必备	FID
记录	Record	必备	Record

GetBroadcastData 原语 **表 29**

参数名称	英文表示	请求	确认	ASN.1 型
目录标识	DID	必备	—	Dsrc-DID
文件标识	FID	必备	—	FID
记录标识	RID	必备	—	INTEGER
记录	Record	—	必备	Record

6.4.2.3参数的设定和释义

(1) 目录标识

接收方的 ASN.1 型专用短程通信目录标识。

(2) 文件标识

广播记录所存储文件号，且要求该文件为记录型文件。

(3) 记录标识

广播信息记录标识号。

(4) 记录

广播信息记录内容，可变长度。

6.4.3 协议规程

6.4.3.1RSU：广播群组的传输

RSU 端的 B-KE 应定期使用带有下列设定参数的 T-KE 的 SET.request 服务来传送附件 A 中规定的广播群组：

(1) IID（空缺）；
(2) LID 是 0xFFFFFFFF；
(3) DID；
(4) FID；
(5) Offset 等于 RID；
(6) FileContent 等于 Record；
(7) Mode 等于 FALSE；
(8) FlowControl 等于 1。

6.4.3.2OBU：广播群组的接收

OBU 的 B-KE 应用带有下列参数的 T-KE 的 SET.indication 来接收广播群组：

(1) IID（空缺）；
(2) LID 等于 0xFFFFFFFF；
(3) FID；
(4) Offset 等于 RID；
(5) FileContent 等于 Record；
(6) Mode 等于 FALSE。

OBU 端的 B-KE 应将 B-KE 的新值 Record 加入广播记录缓冲区中。

6.4.3.3RSU：BroadcastData

若广播记录缓冲区的上条记录内容完全相同，则忽略之；否则 B-KE 应向广播记录缓冲区的内容加入新记录，且当前记录 RID 为 0，以前记录 RID 逐次加 1。

6.4.3.4OBU: GetBroadcastData

B-KE 应使用 GetBroadcastData.request 中给定的 RID 对文件进行获取，并用 GetBroadcastData.confirm 将获得结果交给有 DID 的应用。

6.5 数据结构

6.5.1 模块的使用

T-KE 应使用 DSRCData 模块和 DSRCtransferData 模块。

注 1：IMPORT 和 EXPORT 机制在 GB/T 16262 中作了标准化。

注 2：本附件规定了数据结构。在使用本附件时，使用者应按相应的系统规定数据结构的详细内容，按规定数据结构在 T-KE 中使用 DSRCData 模块和 DSRCtransferData 模块。

6.5.2 ASN.1 模块

```
DSRCData  DEFINITIONS::= BEGIN
    IMPORTS
    ContainerJ-y    FROM ApplicationJ          --此行应对定义容器类型数
                                               据的每个应用者给出，--J
                                               和y应由无歧义的后缀取代。
    RecordJ-y     FROM  ApplicationJ；         --此行应对定义记录类型数
                                               据的每个应用者给出， --J
                                               和y应由无歧义的后缀取代。
    -- EXPORTS everything；

Action-Request::=SEQUENCE{
    mode              BOOLEAN,
    did                   Dsrc-DID,
    actionType            ActionType,
    accessCredentials   OCTET STRING (SIZE(0..127,...)) OPTIONAL,
    actionParameter       Container OPTIONAL,
    iid                 Dsrc-DID OPTIONAL
    }

Action-Response::=SEQUENCE{
    fill                BIT STRING (SIZE(2)),
    did                   Dsrc-DID,
```

```
    responseParameter    Container OPTIONAL,
    iid                 Dsrc-DID OPTIONAL,
    ret                 ReturnStatus
    }

ActionType::=INTEGER(0..127,...)
    -- 在GB/T XXXXX-XXXX（《电子收费  专用短程通信  设备应用》）中已经
定义如下操作：
    --  0  getSecure
    --  1  setSecure
    --  2  getRand
    --  3  transferChannel
    --  4  setMMI
    --  (5.. 80)    保留给DSRC应用
--  (81..127)  保留给私有应用

ApplicationContextMark::=Container
       (WITH COMPONENTS {octetstring  PRENSENT})
    -- ApplicationContextMark的示例可在第七章中找到，参考SysInfoFile
的相关内容。

ApplicationList::=SEQUENCE (SIZE (0..127,...)) OF
    SEQUENCE{
       aid                      DSRCApplicationEntityID,
       did                   Dsrc-DID OPTIONAL,
       applicationParameter    ApplicationContextMark OPTIONAL
       }

BeaconID::= SEQUENCE{
    manufacturerID       INTEGER(0..255),
    individuaLID         INTEGER(0..16777215)
}

BroadCastPool::=File

BST::=SEQUENCE{
    fill                      BIT STRING(SIZE(3)),
    rsu                    BeaconID,
    time                   Time,
    profile                   Profile,
    mandapplications          ApplicationList,
    nonmandapplications       ApplicationList OPTIONAL,
    profileList                SEQUENCE (0..127,...) OF Profile
```

```
    }

Container::=CHOICE{
    integer                  [0] INTEGER,
    bitstring                    [1] BIT STRING,
    octetstring              [2] OCTET STRING(SIZE(0..127),...),,
    universalString          [3] UniversalString,
    beaconID                 [4] BeaconID,
    t-apdu                       [5] T-APDUs,
    dsrcApplicationEntityId      [6] DSRCApplicationEntityID,
    dsrc-Ase-Id              [7] Dsrc-DID,
    fileId                   [8] FID,
    file                     [9] File,
    broadcastPool            [10] BroadcastPool,
    directory                [11] Directory,
    time                     [12] Time,
    vector                       [13] SEQUENCE (0..255) OF
INTEGER(0..127,...),
    filelist                     [14] FileList,
    dummy                        [15..96]   -- 保留为DSRC应用,
    private                    [97..127]   -- 保留给私有应用,
     ...
     , contI-x                   [i]    ContainerI-x  -- 此行应对每个
进口ContainerI.x给出,
--其中I.x 被相关的后缀取代， i 为从0
                                                --开始的已登记的标签。
                                                -- 间隙应由contI.x [i] BIT
STRING来填充;
 }

Create-Request::=SEQUENCE{
    fill                BIT STRING (SIZE(2)),
    did                  Dsrc-DID,
    accessCredentials    OCTET STRING(SIZE(0..127,...)) OPTIONAL,
    filelist            FileList,
    iid                  DsrcApplicationEntityID OPTIONAL
    }

Create-Response::=SEQUENCE{
    fill              BIT STRING(SIZE(3)),
    did               Dsrc-DID,
    iid               Dsrc-DID OPTIONAL,
     ret               ReturnStatus
    }
```

```
Directory::=SEQUENCE (SIZE(0..127, ...)) OF FID

Dsrc-DID::=INTEGER(0..127,...)    -- DirectroyID

DsrcApplicationEntityID::=INTEGER{
    system                              (0),
    electronic-toll-collection          (1),  -- 电子应用
    road-tag-station-management         (2),  -- 道路标识站应用
    electronic-road-price               (3),   --城市道路收费
    freight-fleet-management            (4),
    public-transport                    (5),
    traffic-traveller-information       (6),
    traffic-control                       (7),
    parking-management                  (8),
    geographic-road-database            (9),
    medium-range-preinformation         (10),
    man-manchine-interface              (11),
    emergency-warning                   (12)
    } (0..31,...)
     -- (17－28) 保留给DSRC应用
    -- (28－31) 保留给私有应用

Event-Report-Request::=SEQUENCE{
    mode                 BOOLEAN,
    did                  DirectoryID,
    eventType           EventType,
    accessCredentials   OCTET STRING (SIZE(0..127,...)) OPTIONAL,
    eventParameter      Container OPTIONAL,
    iid                 Dsrc-DID OPTIONAL
    }

Event-Report-Reponse::=SEQUENCE{
    fill                BIT STRING (SIZE(2)),
    did                  DirectoryID,
    iid                 Dsrc-DID OPTIONAL,
    ret                 ReturnStatus OPTIONAL
    }

EventType::=INTEGER{
    release             (0)
    } (0..127,...)
     --  (1－80)保留为DSRC应用
    --  (81-127)保留为自用
```

```
FID::= INTEGER(0..127,...)

File::= OCTET STRING(SIZE(0..127, ...))

FileInfo::=SEQUENCE{
    fileID            FID,
     length               INTEGER(0..32767, ...),
    type              FileType,               -- 文件类型
    accessRights      FileAccessRights         --访问权限
}

FileAccessRights::= SEQUENCE{
    read              BIT STRING (SIZE (2)),
    write           BIT STRING (SIZE (2)),
    reserved        BIT STRING (SIZE(4))
    }

FileIdList::=SEQUENCE(SIZE(0..127, ...))OF FID

FileList::=SEQUENCE(SIZE(0..127, ...)) OF FileInfo

FileType::=INTEGER{
    typeBinary        (0),
    typeRecord        (1)
}(0..127, ...)

Get-Request::=SEQUENCE{
    fill              BIT STRING (SIZE(2)),
    did               Dsrc-DID,
    accessCredentials    OCTET STRING(SIZE(0..127,...)) OPTIONAL,
    iid                  DsrcApplicationEntityID OPTIONAL,
    fileID               FID,
    offset               INTEGER(0..32767,...),
    length               INTEGER(0..127,...)
    }

Get-Response::=SEQUENCE{
    fill                BIT STRING (SIZE(2)),
    did                  Dsrc-DID,
    iid                 Dsrc-DID OPTIONAL,
    fileId              FID,
    fileContent         File OPTIONAL,
    ret                 ReturnStatus
```

```
    }

Initialization-Request::=BST

Initialization-Response::=VST

NamedFile::=SEQUENCE{
    fileID                  FID,
    fileContent             File
}

ObuStatus::=SEQUENCE{
    iccPresent        BOOLEAN,    -- 存在（0），无（1）
     iccType              BIT STRING (SIZE(3)),    -- 逻辑加密接触卡
 （X1），CPU接触卡（X0）
                                                    --逻辑加密非接触卡
 （1X），CPU接触卡（0X）
    iccStatus         BOOLEAN,    -- IC卡正常（0），出错（1）
    locked              BOOLEAN,    -- OBU未锁（0），被锁（1）
    tampered          BOOLEAN,    -- OBU未被拆动（0），被拆动（1）
    battery             BOOLEAN,    -- OBU电池正常（0），电池电量低（1）
    reservedBits      BIT STRING (SIZE(8))
    }

ObuConfiguration::=SEQUENCE{
    macID                INTEGER(0..4294967295), -- MAC地址,同时作为
LID。
    equipmentClass       BIT STRING (SIZE(4)),    -- 0 不支持IC卡接口  1
支持IC卡接口
    equipmentVersion    BIT STRING (SIZE(4)),
    obuStatus           ObuStatus
    }

Profile::=INTEGER(0..127, ...)
     -- b6b5b4 表示配置号；b3b2b1b0 表示配置的所支持的信道号
     -- 00H 配置0（A类）的信道1
     -- 01H 配置0（A类）的信道2
     -- 10H 配置1（B类）的信道1
     -- 11H 配置1（B类）的信道2
    -- (0一80) 保留给DSRC应用
    -- (81-127) 保留给私有应用

Record::=SEQUENCE OF Container
```

```
Remove-Request::=SEQUENCE{
    fill                BIT STRING (SIZE(2)),
    did                  Dsrc-DID,
    accessCredentials    OCTET STRING(SIZE(0..127,...)) OPTIONAL,
    fileIdlist          FileIdList,
    iid                  DsrcApplicationEntityID OPTIONAL
}

Remove-Response::=SEQUENCE{
    fill                 BIT STRING(SIZE(3)),
    did              Dsrc-DID,
    iid              Dsrc-DID OPTIONAL,
     ret                 ReturnStatus
    }

ReturnStatus::=INTEGER{
    noError                 (0),
    accessDenied         (1),
    argumentError        (2),
    complexityLimitation (3),
    processingFailure    (4),
    processing            (5),
    chainingError        (6)
    }(0..127,...)
     --  (7一127) 保留给DSRC应用

RID::= INTEGER(0..127,...)

Set-Request::=SEQUENCE{
    fill                BIT STRING (SIZE(1)),
    mode                 BOOLEAN,
    did                 Dsrc-DID,
    accessCredentials   OCTET STRING(SIZE(0..127,...)) OPTIONAL,
    iid                 Dsrc-DID OPTIONAL,
    fileId              FID,
    offset              INTEGER(0..32767,...),
    length               INTEGER(0..127,...),
    fileContent         File
    }

Set-Response::=SEQUENCE{
    fill                BIT STRING(SIZE(3)),
    did                  Dsrc-DID,
    fileId              FID,
```

```
    iid                  Dsrc-DID OPTIONAL,
     ret                 ReturnStatus
    }

Time::=INTEGER(0..424967295)-- 从1970年1月1日00:00(协调世界时)起，以
秒计算的累计总时间。

T-APDUs::= CHOICE{
    action-request           [0] Action-Request,
    action-response          [1] Action-Response,
    create-request           [2] Create-Request,
    create-response      [3] Create-Response,
    remove-request           [4] Remove-Request,
    remove-response      [5] Remove-Response,
    event-report-request     [6] Event-Report-Request,
    event-report-response    [7] Event-Report-Response,
    set-request          [8] Set-Request,
    set-response         [9] Set-Response,
    get-request          [10] Get-Request,
    get-response         [11] Get-Response,
    initialization-request   [12] Initialization-Request,
    initialization-response [13] Initialization-Response
 }

VST::=SEQUENCE{
    fill                     BIT STRING (SIZE(4)),
    profile                  Profile,
    applications          ApplicationList,
    obuConfiguration     ObuConfiguration
    }

END

DSRCtransferData  DEFINITIONS::= BEGIN
 IMPORTS  T-APDUs  FROM  DSRCData；
 -- EXPORTS everything;
Message::=  T-APDUs  --消息在DSRC 链路上传输;

END
```

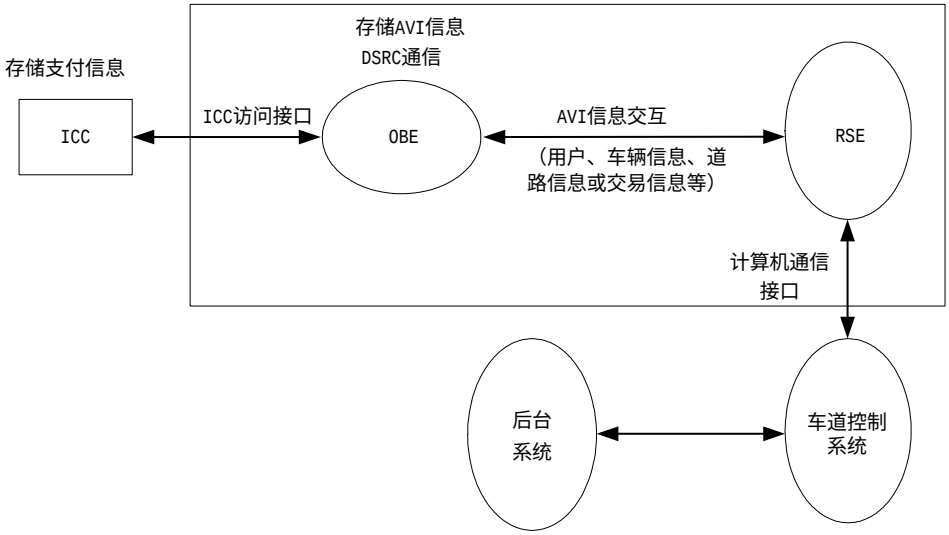
7 设备应用

7.1 应用总则

7.1.1 ETC 系统构成

ETC 系统由前端系统和后台系统组成，前端系统包括车道控制系统、RSE、OBE 以及 ICC。

OBE 应为两片式类型，即应支持 ICC 的读写。在 ETC 应用中，涉及电子支付的功能应由 ICC 实现，OBE 提供 ICC 至 RSE 信息转发功能。系统构成见图 25。



注：方框中的内容为本标准所涉及的内容。

图25 电子收费系统构成

典型ETC交易示例见图26。

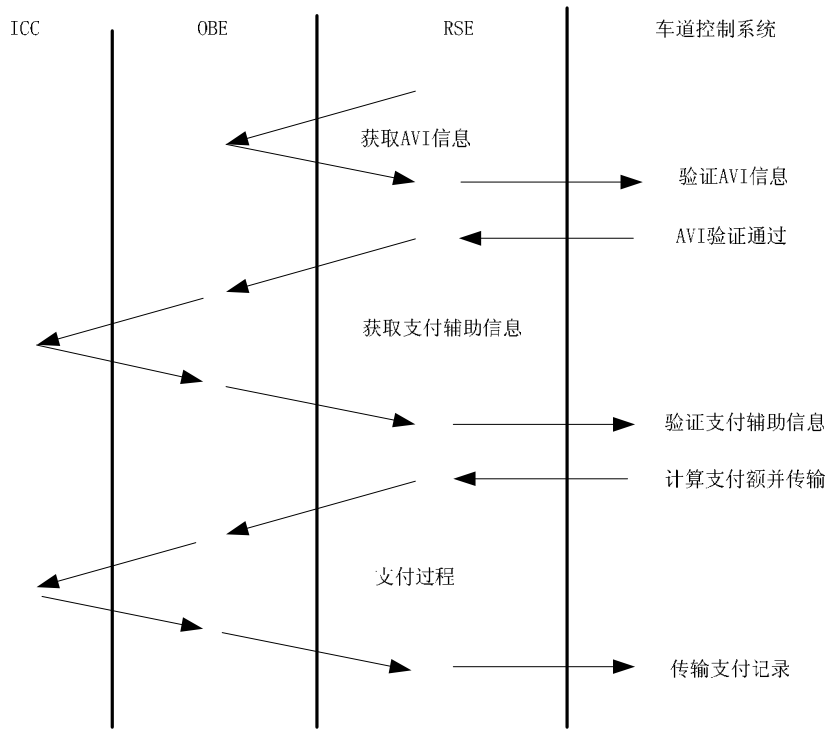


图26　ETC 交易示例

7.1.2 OBE 数据一般规定

OBE内的数据采用目录与文件进行组织。

OBE内目录分为两类：系统目录及应用目录，系统目录为OBE的根目录，只有一个；应用目录是根目录下的子目录，可有多个，一个应用对应一个目录。每个目录下有多个文件，子目录下不应有其他子目录。

文件分为密钥文件和应用文件两类：密钥文件存储用以控制应用数据的安全访问的密钥；应用文件用于存储应用数据。

目录和文件分别以DID和FID来标识。DID编号范围为0x00－0x0F，其中DID为0x00者为根目录。文件编号FID范围为0x00－0x7F，其中FID为0x00号者为密钥文件。根目录下的文件为系统文件，应用文件存储于应用目录中。对于ETC应用，目录号应为0x01。

OBE 内的数据组织结构见表 30。

OBE 数据组织结构表　　**表 30**

目录/文件			文件类型	ASN.1 数据结构 [a]
0x00 根目录	0x00　系统密钥文件		记录	SysKeyFile
	0x01　系统信息文件		二进制	SysInfoFile
	…		…	…
	0x01　ETC 应用目录	0x00　ETC 应用密钥文件	记录	EtcAppKeyFile
		0x01　ETC 应用车辆信息文件	二进制	EtcAppVehicleFile
		…	…	…
	… 其它应用目录	…	…	…
a ASN.1 数据结构见 7.6。				

7.1.3 OBE 密钥

密钥分为四类：主控密钥、维护密钥、认证密钥和加密密钥。主控密钥分系统主控密钥和应用主控密钥，系统主控密钥在系统密钥文件中，应用主控密钥在对应的应用密钥文件中。密钥的功能见表 31。

密钥功能 **表 31**

密钥类型	密钥代码	密钥功能	应用过程
主控密钥	0xM0	(1) 认证通过后可创建文件； (2) 对自身进行安全写入； (3) 对同目录下其它密钥的安全写入； (4) 根目录下的主控密钥用于子目录主控密钥的写入	发行
维护密钥	0xM1	本目录文件的安全方式写入	发行
认证密钥	0xM2	本目录文件访问读写权限控制	交易
加密密钥	0xM3	传输过程中数据的加密与解密处理	交易
注：M 为密钥所属的目录编号，如 ETC 应用的主控密钥为 0x10。			

7.1.4 文件属性

目录和文件操作由密钥控制。

目录和文件的删除与创建在主控密钥的控制下进行。

普通文件（非密钥文件）有四种属性，见表32。

文件属性 **表 32**

属性	代码	描述
自由	0x00	自由读或写，不需要任何认证或加密处理，传输为明文数据
认证	0x01	认证后可以读或写，传输为明文数据
加密	0x02	读或写的数据内容进行加密处理，传输为密文数据
无权限	0x03	无读或者写的权限

7.1.5 扩展应用接口

ETC设备应提供基于ACTION服务原语（见第6节）所扩展的应用接口，见表33。

扩展 ETC 应用接口 **表 33**

操作类型(ActionType)	操作名称	描述
0	GetSecure	安全文件读取，提供MAC和安全加密接口
1	SetSecure	安全文件写入，提供MAC和安全加密接口
2	GetRand	取随机数，用于安全用途
3	TransferChannel	通道传输，用于向OBE部件传输APDU
4	SetMMI	设置人机界面，规范OBE需统一指示的内容

7.1.6 安全

所有的加密和认证过程均通过 PSAM 的方式进行。

储值卡的交易流程应符合 JR/T 0025，交易时间应不大于 270ms。

7.2 关键设备总体技术要求

7.2.1 OBE 总体技术要求

7.2.1.1无线链路通信

OBE 和 RSE 之间的 DSRC 应符合 4、5、6 节的规定。

OBE 的发射机应能够工作在非调制状态，即载波状态。

OBE 的发射机应能够工作在连续发射周期为 511 比特的伪随机二进制序列（PN9）的状态。

7.2.1.2安全

OBE 应提供安全访问模块或者达到同样安全等级的芯片，以存放访问控制密钥和 ETC 应用信息等。

OBE 应支持 TDES 算法的数据存取和访问控制。

OBE 中所有初始化数据的写入应采用 TDES 加密方式传输。

OBE 应具备 ICC 读写接口，该接口应符合 ISO/IEC 7816 或 ISO/IEC 14443 TYPE-A 标准的相关规定。ICC 接触式接口通信速率应不低于 56kbps。

7.2.1.3信息存储

OBE 内的用户信息存储宜采用数据块的方式，寻址应采用目录和文件的方式。

OBE 内应具有不小于 1k 字节作为用户自定义的应用信息存储空间。

7.2.1.4部件

（1）标准配置部件

OBE 应配置的外部部件：ICC 读写接口、字符显示器、蜂鸣器。

（2）可选配置部件

OBE 可选的外部部件：扬声器、指示灯、USB 接口、RS232 串口等有线接口。

7.2.1.5防拆卸与恢复

OBE 应具备防止用户拆卸功能，一旦被拆卸，应当立即在 OBE 内的相应信息存储区中设置相应标志字节/标志位。

因拆卸而引起的 ETC 应用失效应能够通过软件设置的方式得到恢复。

7.2.1.6应用的更新

OBE 应支持应用更新，更新可采用 DSRC 方式或有线方式。

7.2.1.7可靠性

OBE 平均无故障时间应大于 50,000h。

7.2.1.8平均免维护时间

OBE 平均免维护时间不小于 2 年（按每天 10 次交易计算）。

7.2.1.9环境条件

环境条件应符合：

（1） 工作温度： 一般要求-25℃～+70℃（寒区-40℃～+70℃）；
（2） 存储温度：-40℃～+70℃；
（3） 相对工作湿度：5%～100%；
（4） 静电：8kV；
（5） 振动：应符合 GB/T 2423.13；
（6） 冲击：应符合 GB/T 2423.6 试验 Eb 和导则。

7.2.2 RSE 总体技术要求

7.2.2.1工作方式

RSE 采用联机工作方式。
RSE 应提供应用层服务原语接口。

7.2.2.2无线链路通信

RSE 和 OBE 之间的 DSRC 应符合 4、5、6 节的要求。
RSE 的收发天线应具有同轴射频接口。
RSE 的发射机应能够工作在非调制状态，即载波状态。
RSE 的发射机应能够工作在连续发射周期为 511 比特的伪随机二进制序列（PN9）的状态。

7.2.2.3接口

RSE 应至少具有 RS232、RS485、USB 或以太网方式之一的上位机通信接口。
RSE 应具有符合 ISO/IEC 7816 要求的 PSAM 卡座接口，支持对符合 JR/T 0025 安全交易规范要求的 PSAM 的透明指令操作。PSAM 卡通信速率不低于 56kbps。
RSE 宜具有 TTL 电平的光电隔离接口。

7.2.2.4应用的更新

RSE应具有通过上位机接口进行在线应用更新的能力。

7.2.2.5安装

固定安装方式的RSE设备支持户外安装，防护等级应满足GB 4203的要求，并可采用路侧或者顶挂方式；宜采用顶挂安装方式，且吊装在车道正中，挂装高度不低于5.5m。

7.2.2.6通信区域

RSE设备通信区域宽度应可调整在3.3m范围内，见图27。

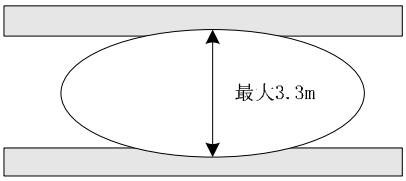


图27　通信区域宽度示意图

7.2.2.7供电

RSE设备应采用220V/50Hz交流电源供电。

7.2.2.8可靠性

RSE 平均无故障时间应大于 70,000h。

7.2.2.9环境条件

环境条件应符合：

(1)　工作温度：一般要求-20℃～+55℃（寒区-35℃～+40℃）；
(2)　存储温度：-40℃～+85℃；
(3)　相对工作湿度：4%～100%；
(4)　静电：8kV；
(5)　振动：应符合 GB/T 2423.13；
(6)　冲击：应符合 GB/T 2423.6 试验 Eb 和导则；
(7)　盐雾：应符合 GB/T 2423.18；
(8)　雷击：抗 4kV 10/200μs 雷击。

7.3　OBE 数据结构

7.3.1 数据结构与属性

OBE的数据分为系统数据和应用数据两类，数据结构与属性见表34。

OBE 数据结构与属性　　**表 34**

目录号	文件号	文件名称	ASN.1 数据结构定义	类型	读属性	写属性
0	0	系统密钥文件	SysKeyFile	记录	0x03	0x01
	1	系统信息文件	SysInfoFile	二进制	0x00	0x02
1	0	ETC 应用密钥文件	EtcKeyFile	记录	0x03	0x01
	1	ETC 应用车辆信息文件	EtcVehicleFile	二进制	0x01	0x02
	2	ETC 应用交易记录文件	EtcTransactionFile	记录	0x00	0x00
	3	ETC 应用保留文件	EtcReservedFile	二进制	0x00	0x00
2	0	标识站应用文件	FlagStationFile	二进制	0x00	0x00

3	0	城市道路收费应用文件	ErpAppFile	二进制	0x00	0x01
…	…	…	…	…	…	…
15	0	保留	…	…	…	…

7.3.2 系统数据

7.3.2.1系统密钥文件

系统密钥文件的目录号为0，文件号为0，ASN.1定义为SysKeyFile，系统密钥文件存储OBE系统信息安全访问控制密钥，文件内容见表35。

系统密钥文件 **表 35**

序号	字段名称	ASN.1 类型	字段内容	密钥代码
1	sysMasterKey	Key	系统主控密钥	0x00
2	sysMaintainKey	Key	系统维护密钥	0x01

7.3.2.2系统信息文件

系统信息文件的目录号为0，文件号为1，ASN.1定义为SysInfoFile，存储OBE发行及合同相关信息，文件内容见表36。

系统信息文件 **表 36**

序号	字段名称	ASN.1 类型	字段内容
1	contractProvider	OCTET STRING (SIZE(8))	服务提供商名称
2	contractType	INTEGER(0..127, ...)	协约类型
3	contractVersion	INTEGER(0..127, ...)	合同版本
4	contractSerialNumber	ContractSerialNumber	合同序列号
5	contractSignedDate	Date	合同签署日期
6	contractExpiredDate	Date	合同过期日期
7	Reserved	OCTET STRING (SIZE(64))	预留

7.3.3 ETC 应用数据

7.3.3.1ETC 应用密钥文件

ETC应用密钥文件目录号为1，文件号为0，ASN.1定义为EtcKeyFile，存储OBE的ETC应用数据安全访问控制密钥，文件内容见表37。

ETC 应用密钥文件 **表 37**

序号	字段名称	ASN.1 类型	字段内容	密钥代码
1	etcMasterKey	Key	ETC 应用主控密钥	0x10
2	etcMaintainKey	Key	ETC 应用维护密钥	0x11
3	etcAccessKey	Key	ETC 应用认证密钥	0x12
4	etcEncryptKey	Key	ETC 应用加密密钥	0x13
注：etcAccessKey 用于认证，etcEncryptKey 用于加密和鉴别码计算				

7.3.3.2ETC 应用车辆信息文件

ETC应用车辆信息文件目录号为1,文件号为1,ASN.1定义为EtcVehicleFile，文件内容见表38。

ETC 应用车辆信息文件 **表 38**

序号	字段名称	ASN.1 类型	字段内容
1	vehicleLicencePlateNumber	OCTET STRING (SIZE(12))	车牌号
2	vehicleLicencePlateColor	OCTET STRING (SIZE(1))	车牌颜色
3	vehicleClass	INTEGER(0..127, ...)	车型
4	vehicleUserType	INTEGER(0..127, ...)	车辆用户类别
5	vehicleDimensions	VehicleDimensions	车辆尺寸
6	vehicleWheels	INTEGER(0..127, ...)	车轮数
7	vehicleAxles	INTEGER(0..127, ...)	车轴数
8	vehicleWheelBases	INTEGER(0..65535)	轴距
9	vehicleWeitghtLimits	INTEGER(0..2^{24}-1)	车辆载重/座位数
10	vehicleSpecificInfomation	OCTET STRING(SIZE(16))	车辆特征描述
11	VehicleEngineNumber	OCTET STRING(SIZE(16))	车辆发动机号
12	vehicleReserved	OCTET STRING(SIZE(10))	保留字段

7.3.3.3ETC 应用交易记录文件

ETC应用交易记录文件目录号为1，文件号为2，ASN.1定义为EtcTransactionFile，ETC交易记录空间写满时，记录用先入先出的方式循环进入记录区,文件内容见表39。

ETC 应用交易记录文件 **表 39**

序号	字段名称	ASN.1 类型	字段内容
1	RecordCount	INTEGER(0..127, ...)	记录数
2	Record1Length	INTEGER(0..127, ...)	第一条记录的长度
3	Record1	OCTET STRING (SIZE(Record1Length))	第一条记录内容
4	Record2Length	INTEGER(0..127, ...)	第二条记录的长度
5	Record2	OCTET STRING (SIZE(Record2Length))	第二条记录内容
…	…	…	…
2n	RecordnLength	INTEGER(0..127, ...)	第 n 条记录的长度
2n＋1	Recordn	OCTET STRING (SIZE(RecordnLength))	第 n 条记录内容

7.3.3.4ETC 应用保留文件

ETC应用保留文件目录号为1，文件号为3，ASN.1定义为EtcReservedFile，文件预留为OCTET STRING (SIZE(40))。

7.3.3.5标识站应用文件

标识站应用文件目录号为2，文件号为0，ASN.1定义为FlagStationFile，文件预留为OCTET STRING (SIZE(40))。

7.3.3.6城市道路收费应用文件

城市道路收费应用文件目录号为3，文件号为0，ASN.1定义为ErpAppFile，文件预留为OCTET STRING (SIZE(40))。

7.4 ETC 应用接口

7.4.1 ACTION 服务原语

ETC应用在应用层服务原语ACTION之上扩展出ETC应用接口，其中ACTION服务原语如下：

```
Action-Request::=SEQUENCE{
    mode                BOOLEAN,
    did                 Dsrc-DID,
    actionType          ActionType,
    accessCredentials   OCTET STRING (SIZE(0..127,...)) OPTIONAL,
    actionParameter     Container OPTIONAL,
    iid                  Dsrc-DID OPTIONAL
    }

Action-Response::=SEQUENCE{
    fill                 BIT STRING (SIZE(1)),
    did                  Dsrc-DID,
    responseParameter    Container OPTIONAL,
    iid                  Dsrc-DID OPTIONAL,
    ret                  ReturnStatus
    }

ActionType::=INTEGER(0..127,...)
```

— 已经定义如下操作：

— 0 GetSecure

— 1 SetSecure

— 2 GetRand

— 3 TransferChannel

— 4 SetMMI

ActionType中，5…80 保留给DSRC应用，81…127保留给私有应用。

7.4.2 GetSecure 服务原语

7.4.2.1功能

GetSecure用于实现安全的数据读取，并提供可选的认证、加密和信息鉴别的机制。其中：

(1) 认证：RSE端提供访问凭证，OBE端验证通过后才允许读取；

（2） 加密：OBE端对数据加密后再传输到RSE端，RSE端需对之解密后才可获取原始数据；

（3） 信息鉴别：对所传输的数据进行加密运算产生MAC，随数据后一起传输；RSE端接收到首先对之进行验证,无误后进行后续处理。

本接口中，（1）和（2）为可选项，（3）为必备项；对于无需（2）或（3）的情形，需调用Get服务实现。

上述三者视安全要求可组合运用。

如果同时出现（2）和（3）的情形，MAC应基于加密后的信息计算得到。

7.4.2.2接口

（1）请求（GetSecure.request）

GetSecure.request参数要求见表40。

GetSecure.request 参数要求 **表 40**

参数	取值	参数说明
mode	TRUE	确认模式，需应答
did	Dsrc-DID	ETC应用等于1
actionType	0	等于0
accessCredentials	—	可选

actionParameter	GetSecureRq::=SEQUENCE{
	fill BIT STRING (SIZE(7)),
	fileid FID,
	offset INTEGER(0..65535,...),
	length INTEGER(0..127,...),
	rndRsuForAuthen Rand,
	keyIdForAuthen INTEGER(0..255),
	keyIdForEncrypt INTEGER(0..255) OPTIONAL
	}
		
		文件标识
		操作数据起始位置
		操作数据长度
		产生MAC用随机数
		产生MAC用密钥索引号
		加密密钥索引号

iid	—	不存在

（2）应答（GetSecure.response）

GetSecure.response参数要求见表41。

GetSecure.response 参数要求 **表 41**

参数	取值	参数说明
did	Dsrc-DID	ETC应用对应1

responseParameter	GetSecureRs::=SEQUENCE {
	fileid FID,
	file File,
	Authenticator OCTET STRING (SIZE(8))
	}
		可选
		文件标识
		读取的数据（可能加密）
		鉴别码

iid	—	不存在
ret	ReturnStatus	必备

7.4.3 SetSecure 服务原语

7.4.3.1功能

SetSecure用于实现安全的数据写入，并提供可选的认证、加密和信息鉴别的机制。其中：

（1） 认证：RSE端提供访问凭证，OBE端验证通过后才允许读取；

（2） 加密：OBE端对数据加密后再传输到RSE端，RSE端需对之解密后才可获取原始数据；

（3） 信息鉴别：对所传输的数据进行加密运算产生信息鉴别码，随数据后一起传输；RSE端接收到首先对之进行验证,无误后进行后续处理。

本接口中，（1）和（2）为可选项，（3）为必选项；对于无需（2）或（3）的情形，需调用Set服务实现。

上述三者示安全要求可组合运用。

如果同时出现（2）和（3）的情形，信息鉴别码应基于加密后的信息计算得到。

7.4.3.2接口

（1）请求（SetSecure.request）

SetSecure.request参数要求见表42。

SetSecure.request 参数要求 **表 42**

参数	取值	参数说明
mode	TRUE	确认模式，需应答
did	Dsrc-DID	ETC应用等于1
actionType	1	等于1
accessCredentials	—	可选

actionParameter	SetSecureRq::=SEQUENCE{
	fill BIT STRING (SIZE(7)),
	fileid FID,
	offset INTEGER(0..65535,...),
	length INTEGER(0..127,...),
	file File,
	rndRsuForAuthen Rand,
	keyIdForAuthen INTEGER(0..255),
	keyIdForEncrypt INTEGER(0..255) OPTIONAL
	}
		
		文件标识
		操作数据起始位置
		操作数据长度
		数据内容（可能加密）
		产生MAC用随机数
		产生MAC用密钥索引号
		加密密钥索引号

iid	—	不存在

（2）应答（SetSecure.response）

SetSecure.response参数要求见表43。

SetSecure.response 参数要求 **表 43**

参数	取值	参数说明

did	Dsrc-DID	ETC应用等于1
responseParameter	—	不存在
iid	—	不存在
ret	ReturnStatus	必备

7.4.4 GetRand 服务原语

7.4.4.1功能

GetRand服务用于获取8字节随机数。

7.4.4.2接口

（1）请求（GetRand.request）

GetRand.request参数要求见表44。

GetRand.request参数要求　　表44

参数	取值	参数说明
mode	TRUE	确认模式，需应答
did	Dsrc-DID	ETC应用等于1
actionType	2	等于2
accessCredentials	—	可选
actionParameter	—	不存在
iid	—	不存在

（2）应答（GetRand.response）

GetRand.response参数要求见表45。

GetRand.response 参数要求　　表 45

参数	取值	参数说明
did	Dsrc-DID	ETC应用等于1

responseParameter	GetRandRs::=SEQUENCE {
	rand Rand
	}
		可选
		随机数

iid	—	不存在
ret	ReturnStatus	必备

7.4.5 TransferChannel 服务原语

7.4.5.1功能

为RSE与OBE部件之间（如ICC、显示器、蜂鸣器等）通信提供通道传输功能，OBE充当RSE与外部件之间的转发器。已经定义的外部件通道标识号见表46。

通道标识号定义表　　表 46

通道标识号	名称	说明

(ChannelID)		
0	OBE	OBE本身
1	ICC	ICC
2	SAM	SAM模块
3	DISPLAY	显示器
4	BEEPER	蜂鸣器
5	SPEAKER	扬声器
6	PRINTER	打印设备
7	SERIAL INTERFACE	串行口
8	USB	USB接口
9	PARALLEL INTERFACE	并行口

7.4.5.2接口

(1) 请求 (TransferChannel.request)

TransferChannel.request参数要求见表47。

TransferChannel.request 参数要求 **表 47**

参数	取值	参数说明
mode	TRUE	确认模式，需应答
did	Dsrc-DID	ETC应用等于1
actionType	3	等于3
accessCredentials	—	可选

actionParameter	ChannelRq::=SEQUENCE{
	channelid ChannelID,
	apdu apduList
	}
		
		通道标识号
		通道指令数据

iid	—	不存在

(2) 应答 (TransferChannel.response)

TransferChannel.response参数要求见表48。

TransferChannel.response 参数要求 **表 48**

参数	取值	参数说明
did	Dsrc-DID	ETC应用等于1
responseParameter	ChannelRs::=SEQUENCE { channelid ChannelID, apdu apduList }	可选 通道标识号 通道应答数据
iid	—	不存在
ret	ReturnStatus	必备

7.4.6 SetMMI 服务原语

7.4.6.1功能

用于规范OBE应用人机界面指示的内容，已经要求指示的内容见表49。

MMI 标识定义 **表 49**

指示内容标识	名称	说明
0	ok	交易处理正常完成
1	error	交易处理异常
2	contactOperator	请联系运营商

7.4.6.2接口

（1）请求（SetMMI.request）

SetMMI.request参数要求见表50。

SetMMI.request 参数要求 **表 50**

参数	取值	参数说明
mode	TRUE	确认模式，需应答
did	Dsrc-DID	ETC应用等于1
actionType	4	等于4
accessCredentials	—	不存在

actionParameter	SetMMIRq::=INTEGER{
	ok (0),
	nok (1),
	contactOperator (2)
	}
		
		0：正常
		1：异常
		2：与运营商联系

iid	—	不存在

（2）应答（SetMMI.response）

SetMMI.response参数要求见表51。

SetMMI.response 参数定义 **表 51**

参数	取值	参数说明
did	Dsrc-DID	ETC应用对应1
responseParameter	—	不存在
ret	ReturnStatus	必备

7.5 ETC 应用安全

7.5.1 安全方式

主要安全保护手段有：

（1） 访问许可：访问数据应提供许可凭证，OBE 验证通过后才允许访问；

（2） 信息鉴别：随关键数据一起传送一组鉴别码，RSE 验证后才认为为合

法数据；

(3) 加密保护：在传输过程中对数据进行加密。

7.5.2 访问许可

OBE 访问许可认证通过后，RSE 具备访问权限，访问许可见图 28。过程如下：

(1) RSE 通过 Get 服务取得 ContractSerialNumber，通过 GetRand 服务或直接从 VST 中获取 RndOBU，RndOBU 宜从 VST 中获取；

(2) RSE 利用 MasterAccessKey（主认证密钥，16 字节）和 ContractSerialNumber 分散出临时认证密钥 tmpAccessKey（16 字节），分散算法如下：

tmpAccessKey = TDES(MasterEtcAppAccessKey, ContractSerialNumber)

(3) RSE 利用临时密钥 tmpAccessKey 加密 RandOBU(8 字节)，产生 accessCredentials，算法如下：

accessCredentials＝TDES(tmpAccessKey, RandOBU)

(4) RSE 后续指令携带 accessCredentials，发送到 OBE；

(5) OBE 利用 AccessKey 和 RndOBU 计算出 tmpAccessCredentials，算法同 c)；

(6) OBE 比较 accessCredentials 和 tmpAccessCredentials 是否相等，相等则赋予该 RSE 访问许可权限。

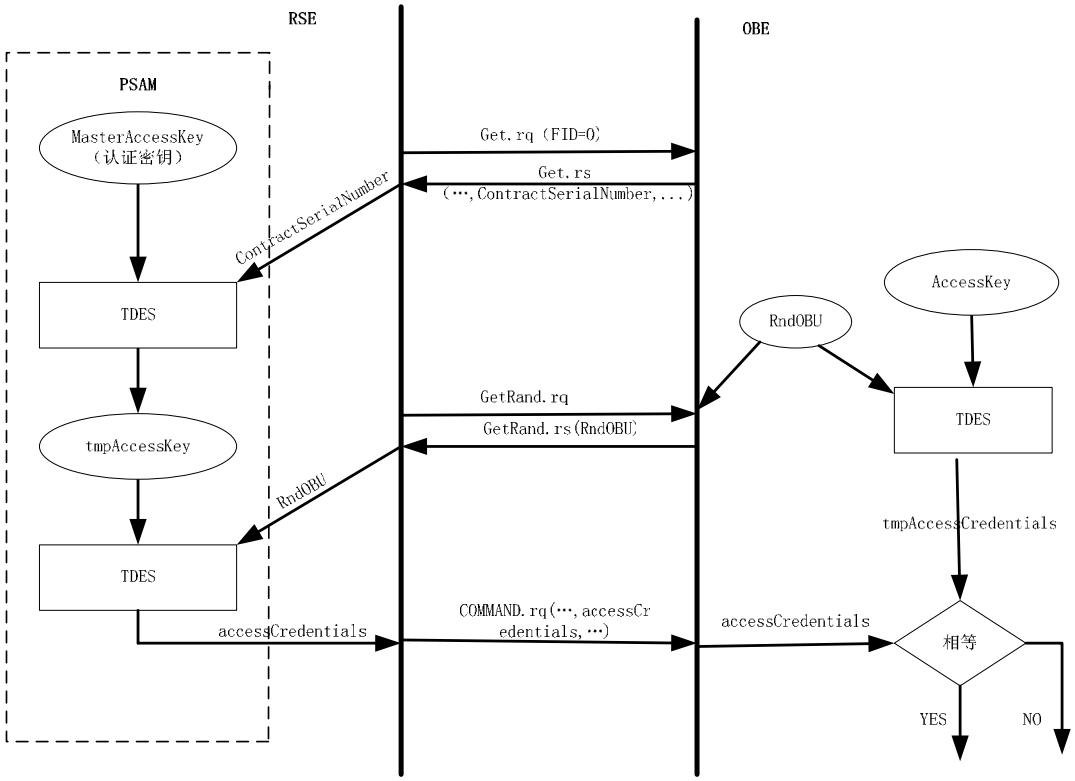


图28 访问许可认证

7.5.3 信息鉴别

信息鉴别主要是针对 OBE 信息的鉴别，以鉴别码的方式实现，信息鉴别见图 29。过程如下：

(1) RSE 通过 Get 服务取得 ContractSerialNumber；

(2) RSE 产生随机数 randRSUForAuthen(8 字节),并随 GetSecure 服务一起发送至 OBE；

(3) OBE 利用 EncryptKey（16 字节）对 randRSUForAuthen、File 内容进行 MAC 计算，得出鉴别码 Authenticator 并随 File 一起作为响应参数发往 RSE。其中 MAC 计算方法如下：

①将 File 内容进行 CRC 计算（多项式 $X^{16}+X^{12}+X^5+1$，起始 FFFFH），产生两字节 CRC0 和 CRC1；

②将 randRSUForAuthen 最低两个字节分别更换为 CRC1，CRC0，形成 8 字节临时数据；

③利用 EncryptKey 对 8 字节数据进行加密计算产生 Authenticator，算法如下：

Authenticator = TDES(EncryptKey, CRC0||CRC1||randRSUForAuthen（高 6 字节）)

(4) OBE 在响应中发送 File 和 Authenticator 至 RSE；

(5) RSE 利用 ContractSerialNumber 和 MasterEncryptKey 计算出临时密钥 tmpEncryptKey（16 字节），算法如下：

tmpEncryptKey = TDES(MasterEtcAppAuthenKey, ContractSerialNumber)

(6) RSE 利用临时密钥 tmpKey、rndRSUForAuthen 及 File，遵循 c）中的算法计算 MAC 码 tmpAuthenticator；

(7) RSE 比较 Authenticator 和 tmpAuthenticator，如果结果相等则为合法数据，否则非法。

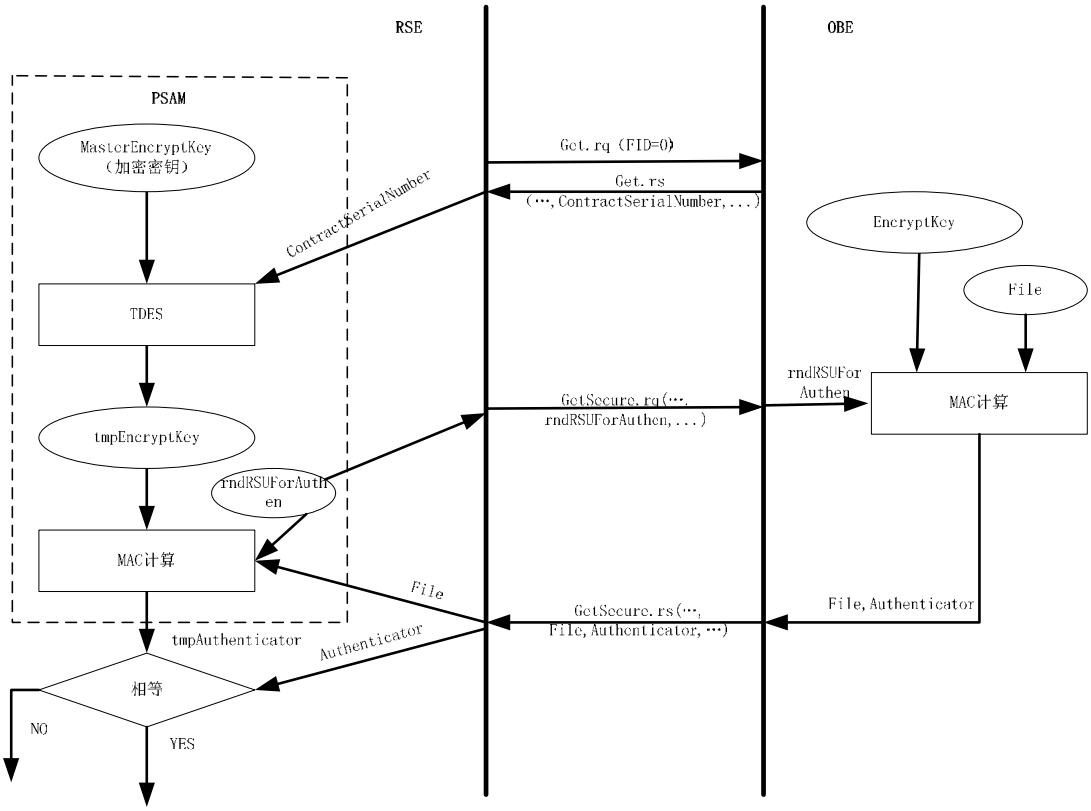


图29　OBE 信息鉴别

7.5.4 获取信息加密

获取 OBE 信息加密算法基于 TDES 实现，见图 30。算法流程如下：

(1)　RSE 发送 GetSecure 服务至 OBE；

(2)　OBE 将文件数据按 8 字节分组，不足则补 0；

(3)　OBE 利用 EncryptKey 对上述结果（8 的整数倍长度）进行 TDES 解密，产生解密结果 decryptFile，算法如下：

$$decryptFile=TDES^{-1}(EncryptKey, Encrypfile)$$

(4)　OBE 将 decryptFile 随 GetSecure.rs 发送至 RSE；

(5)　RSE 利用 MasterEncryptKey 和 ContractSerialNumber 产生临时加密密钥 tmpEncryptKey，算法如下：

$$tmpEncryptKey=TDES(MasterEncryptKey, ContractSerialNumber)$$

(6)　RSE 利用 tmpEncryptKey 对 decryptFile 加密，结果去掉多余 0 字节后即为所读取数据内容。

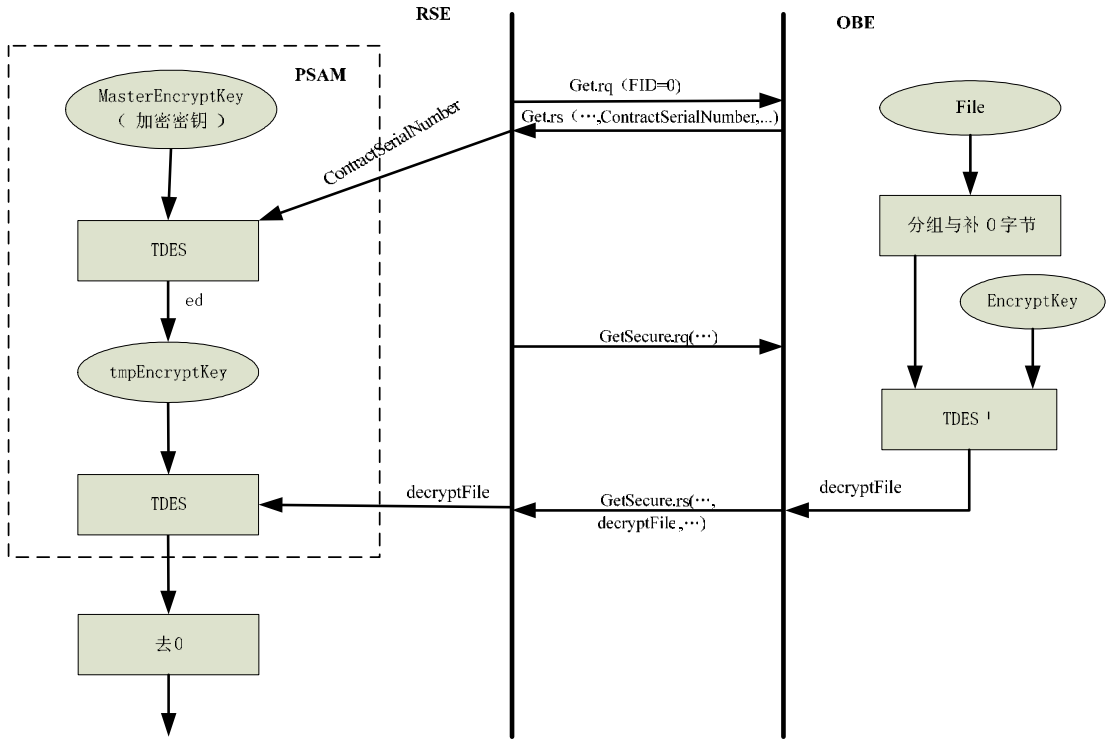


图30　获取信息加密

7.5.5 写入信息加密

写入信息加密算法基于 TDES 实现，见图 31。算法流程如下：

(1)　RSE 将文件数据按 8 字节分组，不足则补 0；

(2)　RSE 利用 MasterEncryptKey 和 ContractSerialNumber 产生临时加密密钥 tmpEncryptKey：

$$tmpEncryptKey=TDES(MasterEncryptKey, ContractSerialNumber)$$

(3)　RSE 利用 tmpEncryptKey 对 (1) 结果加密，加密后的数据 decryptFile 随 SetSecure 服务发送至 OBE；

(4)　OBE 利用 EncryptKey 对 decriptFile（8 的整数倍长度）进行 TDES 解密，产生解密结果 decryptFile：

$$decryptFile=TDES^{-1}(EncryptKey, Encrypfile)$$

(5)　OBE 将解密后的数据去掉多余 0 后写入 OBE 文件，并应答 SetSecure。

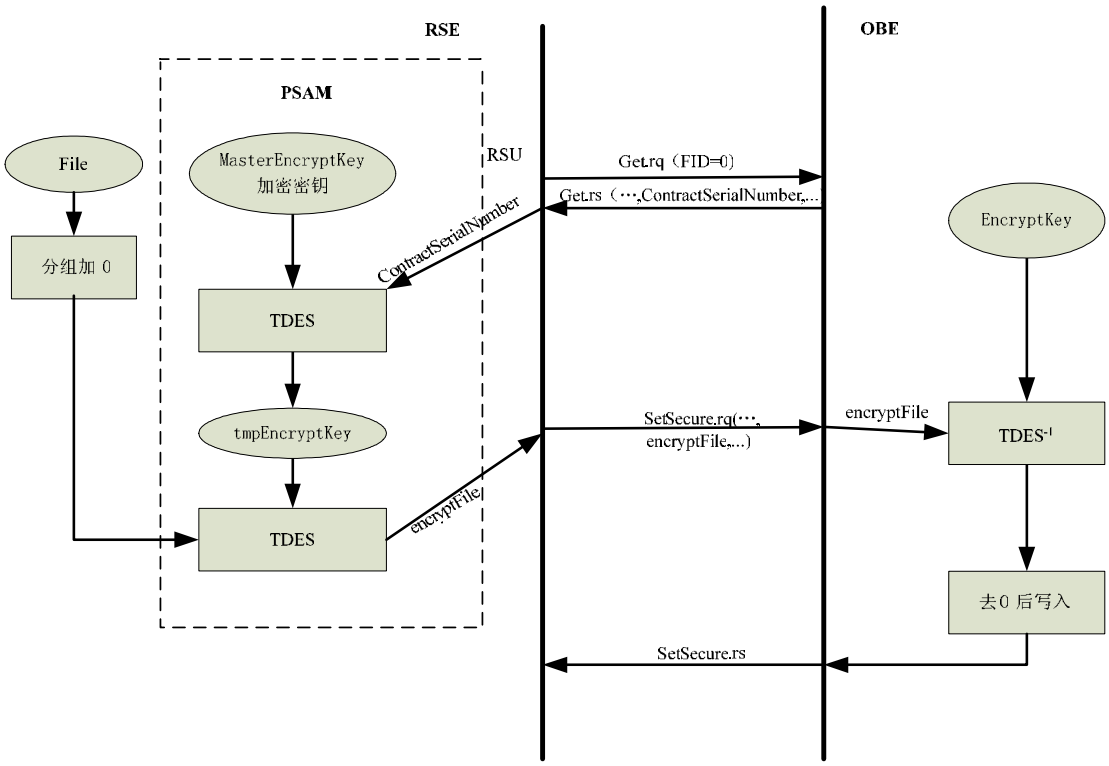


图31　写入信息加密

7.6　OBE 的 ASN.1 型数据结构

```
ETCModule  DEFINITIONS::= BEGIN
  -- EXPORTS everything；
       Container1-0::=[20] GetSecureRq
       Container1-1::=[21] GetSecureRs
       Container1-2::=[22] SetSecureRq
       Container1-3::=[23] SetSecureRs
       Container1-4::=[24] ChannelRq
       Container1-5::=[25] ChannelRs
       Container1-6::=[26] SetMMIRq
       Container1-7::=[27] SetMMIRs
       Container1-8::=[28] Key
       Container1-9::=[29] Rand
       Container1-10::=[30] Date

       Container1-11::=[31] SysKeyFile
       Container1-12::=[32] SysInfoFile
       Container1-13::=[33] EtcKeyFile
       Container1-14::=[34] EtcVehicleFile
       Container1-15::=[35] EtcTransactionFile
       Container1-16::=[36] FlagStationFile
       Container1-17::=[37] ErpAPPFile
```

```
Container1-18::=[38] EtcReservedFile

ChannelID::=INTEGER{
   obe              (0),
   icc              (1),
   sam              (2),
   display          (3),
   beeper           (4),
   printer          (5),
   serialInterface  (6),
   parallelInterface (7)
   }
ApduList::=SEQUENCE OF OCTET STRING(0..127)

ChannelRq::=SEQUENCE{
    channelid          ChannelID,
    apdu                ApduList
   }

ChannelRs::=SEQUENCE {
    channelid          ChannelID,
    apdu               ApduList
   }

ContractSerialNumber::=SEQUENCE{
   contractProviderID           OCTET STRING (SIZE(2)),
   contractInividualID          OCTET STRING (SIZE(6))
   }
 -- 由统一机构为contractProviderID编码

Date::=SEQUENCE{
   year    OCTET STRING(SIZE(2)),    -- BCD编码，YYYY
   month   OCTET STRING(SIZE(1)),    -- BCD编码，MM
   day     OCTET STRING(SIZE(1))     -- BCD编码，DD
   }

EtcKeyFile::=SEQUENCE{
    etcMasterKey              Key,    -- 16字节密钥
    etcMaintainKey            Key,
   etcAccessKey              Key,
   etcEncryptKey             Key
   }
```

```
FlagStationFile::=OCTET STRING (SIZE(40))   --具体内容由标识站应用定义
EtcReservedFile::=OCTET STRING (SIZE(40))  --备ETC系统扩展用途
EtcTransactionFile::=SEQUENCE OF Record

EtcVehicleFile::=SEQUENCE{
   vehicleLicencePlateNumber    OCTET STRING (SIZE(12)),
   vehicleLicencePlateColor     OCTET STRING (SIZE(2)),
   vehicleClass                 INTEGER(0..127,...),
    vehicleUserType              INTEGER(0..127,...),
   vehicleDimensions            VehicleDimensions,
   vehicleWheels                INTEGER(0..127,...),
   vehicleAxles                 INTEGER(0..127,...),
    vehicleWheelBases            INTEGER(0..65535),
   vehicleWeightLimits          INTEGER(0..16777215),
   vehicleSpecificInfomation    OCTET STRING (SIZE(16)),
    vehicleEngineNumber          OCTET STRING(SIZE(16)),
    vehicleReserved              OCTET STRING(SIZE(10))
}
-- vehicleLicencePlateNumber：车牌号码，全牌照（汉字＋字母＋数字）信息，采用字符--型存贮，汉字采用GB2312码，如：“ 京 ”编码为“ BEA9 ”。
-- vehicleLicencePlateColor：车牌颜色，二进制编码表示（0－蓝色，1－黄色，2－黑色， --3－白）。
-- vehicleClass：车辆类型，1字节，1-一型车；2-二型车；3-三型车；4-四型车；5-五型--车；6-六型车；7～10：自定义；11～20：用于计重收费货车车型分类。其中，11-一型--车；12-二型车；13-三型车；14-四型车；15-五型车；16-六型车；17～20：自定义计--重货车车型；21～--50：自定义；50～255：保留给未来使用。
-- vehicleUserType: 车辆用户类型，1字节，0-普通车；6-公务车；8-军警车；10-紧急车；--12-免费；14-车队；0～20内其他：自定义；21～255：保留给未来使用。
-- vehicleDimensions: 车辆尺寸，二进制分别表示长（2字节）、宽（1字节）、高（1字--节）。单位为分米。如0x012C、0x28、0x1E表示30米长、4米高、3米宽。
-- vehicleWheels: 车轮数，二进制表示的数目
-- vehicleAxles: 车轴数，二进制表示的数目。
-- vehicleWheelbases: 轴距，二进制表示，长度为2个字节，单位为分米。如0x28，表示--轴距为4米。
-- vehicleWeightLimits: 车辆载重（货车）或座位数（客车），二进制表示，单位为公斤--/座。
-- vehicleSpecificInfomation: 车辆特征描述，字符用ASCII编码表示，汉字用机内码表--示，如“ 奔驰307 ”。
--vehicleEngineNumber：车辆发动机号。
```

```
    -- VehicleReserved： 保留
ErpAppFile::=OCTET STRING (SIZE(40)) -- ERP应用自定义

    GetRandRs::=SEQUENCE {
        rand      Rand      -- 8字节随机数
       }

   GetSecureRq::=SEQUENCE{
        fill                BIT STRING (SIZE(7)),
        fileid              FID,
        offset              INTEGER(0..65535,...),
        length              INTEGER(0..127,...),
        rndRsuForAuthen     Rand,
        keyIdForAuthen      INTEGER(0..255),
        keyIdForEncrypt      INTEGER(0..255) OPTIONAL   --如果不选
   表示不需对数据加密
        }

   GetSecureRs::=SEQUENCE {
        fileid              FID,
        file                File,
        authenticator       OCTET STRING (SIZE(8))
        }

    Key::=OCTET STRING (SIZE(16))

    Rand::=OCTET STRING (SIZE(8))

    SetMMIRq::=INTEGER{
        ok                (0),   --交易正常
        nok               (1),    --交易异常（通信、设备故障等技术
方面异常）
        contactOperator   (2)    --联系运营商（过期、黑名单等管理
方面异常）
       }

SetSecureRq::=SEQUENCE{
        fill                BIT STRING (SIZE(7)),
       fileid              FID,
       offset              INTEGER(0..65535,...),
       length              INTEGER(0..127,...),
       file                File,
       rndRsuForAuthen     Rand,
       keyIdForAuthen      INTEGER(0..255),
```

```
    keyIdForEncrypt      INTEGER(0..255) OPTIONAL    -- 如果不选，表示数据没有加密
    }

    SysInfoFile::= SEQUENCE{
        contractProvider            OCTET STRING (SIZE(8)),
        contractType                INTEGER(0..127,...),
        contractVersion             INTEGER(0..127,...),
        contractSerialNumber        ContractSerialNumber,
        contractSignedDate          Date,
        contractExpiredDate         Date,
        reserved                    OCTET STRING (SIZE(64))
    }
    -- contractProvider：服务提供商，ASCII编码，服务提供商汉字简单描述，如“华北高速”
    -- contractType：服务类型，服务提供商所提供的服务种类，由各联网收费区域自定义（如--不同费率定义等）
    -- contractVersion：服务版本，服务提供商提供的服务版本，由各联网收费区域自定义
    -- contractSerialNumber：服务序列号，由服务商编号和个体序列号组成
    -- contractSignedDate：合同签署生效日期，BCD编码 YYYYMMDD
    -- contractExpiredDate：合同过期日期，BCD编码，YYYYMMDD

    SysKeyFile::=SEQUENCE{
        sysMasterKey            Key,
        sysMaintainKey          Key
        }

    VehicleDimensions::=SEQUENCE{
        vehicleLength               INTEGER(0..65535),
        vehicleWidth                INTEGER(0..255),
        vehicleHeigth               INTEGER(0..255)
        }
END
```