ENGLISH TRANSLATION

DSRC APPLICATION SUB-LAYER

ARIB STANDARD

VERSION 1.0

ARIB STD－T88

Version 1.0 MAY 25, 2004

Association of Radio Industries and Businesses (ARIB)

General Notes to the English translation of ARIB Standards and Technical Reports

1. The copyright of this document is ascribed to the Association of Radio Industries and Businesses (ARIB).
2. All rights reserved. No part of this document may be reproduced, stored in are retrieval system, or transmitted, in any form or by any means, without the prior written permission of ARIB.
3. The ARIB Standards and ARIB Technical Reports are usually written in Japanese and approved by the ARIB Standard Assembly. This document is a translation into English of the approved document for the purpose of convenience of users. If there are any discrepancies in the content, expressions, etc., between the Japanese original and this translated document, the Japanese original shall prevail.
4. The establishment, revision and abolishment of ARIB Standards and Technical Reports are approved at the ARIB Standard Assembly, which meets several times a year. Approved ARIB Standards and Technical Reports, in their original language, are made publicly available in hard copy, CDs or through web posting, generally in about one month after the date of approval. The original document of this translation may have been further revised and therefore users are encouraged to check the latest version at an appropriate page under the following URL:
   http://www.arib.or.jp/english/index.html
5. The original "DSRC APPLICATION SUB-LAYER (ARIB STD-T88)" is written in Japanese and has been approved by the 53th Standard Assembly Meeting (MAY 25, 2004).
6. The note about IPR (Industrial Property Rights) in the FOREWORD of the standard applies to the use of Essential IPR for the ARIB Standard in Japan. If the ARIB Standard is adopted outside Japan, Essential IPR will be treated in accordance with policies stated by each IPR owner. The IPR owners are, however, expected to apply the rules of the preface of the "Guidelines for Treatment of Industrial Property Rights in connection with the ARIB Standard" (September 5, 1995, approved by the 1st Standard Assembly Meeting). In the preface of the Guidelines, it is stated that it is "desirable that the Essential IPR which relates to any or all parts of the contents of the ARIB Standard should be used free of charge by anyone and that it would not block the use of such Essential IPR in any other country where such an ARIB Standard is adopted."

FOREWORD

The Association of Radio Industries and Businesses (ARIB) has been investigating and summarizing the basic technical requirements for establishing standards. These will appear in the form of standards and specifications governing the use of radio transmission facilities and equipment. The standards are being developed based on the participation of and discussions with, the various radio equipment manufacturers, operators and users.

The standards and specifications contained herein will serve as guidelines for developing standards for private use based on the publicly established technical standards in Japan. Their purpose is to enable effective use of radio frequencies by avoiding interference among users, conflicts among the standards of individual operators, and so forth, so that all parties involved, including radio equipment manufacturers, users and others will be able to ensure the quality and compatibility of radio facilities and equipment.

This standard is being established principally for “DEDICATED SHORT-RANGE COMMUNICATION (DSRC) Application Sub Layer”, which is layered between a DSRC protocol stack, as specified in an ARIB standard STD-T75 “DEDICATED SHORT-RANGE COMMUNICATION (DSRC) SYSTEM” and DSRC Applications for easy deployment of multi application services for the existing DSRC. In order to ensure fairness and openness among all par-ties involved, during drafting stages, we invite radio equipment manufacturers, operators and users both domestically and overseas to participate openly in the activities of the Standard Assembly so as to develop standards with the total agreement of all parties involved.

The scope of application of these standards covers the minimum requirements for communications. They are designed to serve as practical guidelines for operators in developing original specifications and systems that fall within the scope of the standards.

We hope that the standards will aid all parties involved, including radio equipment manufacturers, operators, users, and others in the development of an excellent radio telecommunication system.

NOTE Although this ARIB Standard contains no specific reference to any Essential Industrial Property Right relating thereto, the holder of such Essential Industrial Property Right states that "YYY" is the holder of the Industrial Property Right "XXX" covering this ARIB Standard and to grant a non-exclusive and non-discriminatory license to the use of such right "XXX" on reasonable terms and conditions to anyone using this ARIB Standard. However this does not apply to anyone who uses this ARIB Standard and also owns and lays claim to any other Essential Industrial Property Right whose scope is included in any or all parts of the contents of the provisions of this ARIB Standard.

List of Essential Industrial Property Right

Patent Holder	Name of Patent	Registration No. /Application No.	Remarks
Mitsubishi Electric Corporation	Road to Vehicle communication system	2003-355354	
Hitachi, Ltd.	A comprehensive confirmation form has been submitted with regard to ARIB STD-T88 Ver.1.0		
Motorola Incorporated	A comprehensive confirmation form has been submitted with regard to ARIB STD-T88 Ver.1.0		

Contents

1 General

1.1 Overview

This standard specifies the “DEDICATED SHORT-RANGE COMMUNICATION (DSRC) Application Sub Layer (DSRC-ASL)”, which realizes easy deployment of multi DSRC application services by implementing supplemental DSRC-ASL functions for DSRC protocol stack as specified in an ARIB standard STD-T75 “DEDICATED SHORT-RANGE COMMUNICATION (DSRC) SYSTEM” in a communication between a DSRC mobile stations and a base station.

NOTE This standard originates from the 2nd clause (DSRC-ASL) and related annexes of an ARIB technical report TR-T17 “DEDICATED SHORT-RANGE COMMUNICATION (DSRC) APPLICATION SUB LAYER SPECIFICATION AND TEST ITEMS AND CONDITIONS FOR LAND MOBILE STATION COMPATIBILITY CONFIRMATION”. This standard is also an extended communication control standard on DSRC protocol stacks as specified in the ARIB standard STD-T75.

1.2 Scope of application

This Standard applies to applications based on a DSRC system (hereafter "System"), which consists of a Base Station, a Land Mobile Station (or abbreviated "Mobile Station") and a Test Equipment as specified in the ARIB standard STD-T75.

This Standard specifies the extended communication protocol, which interfaces between DSRC protocol stacks and Network applications or Non-Network applications, and provides complementary communication functions for these DSRC applications.

1.3 Scope of standard

In terms of mutual connectivity and compatibility, this standard defines the minimum level of specifications required for basic connections and services as the mandatory requirement, and the specifications required for what free choice is permitted, such as protocols, as optional standard to provide for future expansion.

Further, in order to provide options and future expansion capabilities as much as possible, care has been taken not to place restrictions on non-standardized specifications.

This standard is applied to DSRC system based on the ARIB STD-T75. Figure1.2 outlines the relationship between ARIB STD-T75 and this standard. The ARIB STD-T75 adopts the 3 Layer structure of the Open Systems Interconnection (OSI) basic reference model and the standardized objects are Layer 1, Layer 2 and Layer 7. This standard defines an application sub layer on that upper layer (Layer 7).

The parameters or information filed as defined “reserve” in this standard are reserved for future expansion capabilities. These parameters or information filed as defined “reserve” in this standard (version) may be defined as specific values or identifiers. The user of this standard should take into account that these values or identifiers might be changed in the

future version.

Parameters specified in this standard are described using Abstract Syntax Notation One :ISO/IEC 8824). The coding rule is a packed encoding rule (UNALIGNED PER (Packed Encoding Rule: ISO/IEC 8825-2). Refer to Annex B for the detail of ASN.1 type parameter specifications.

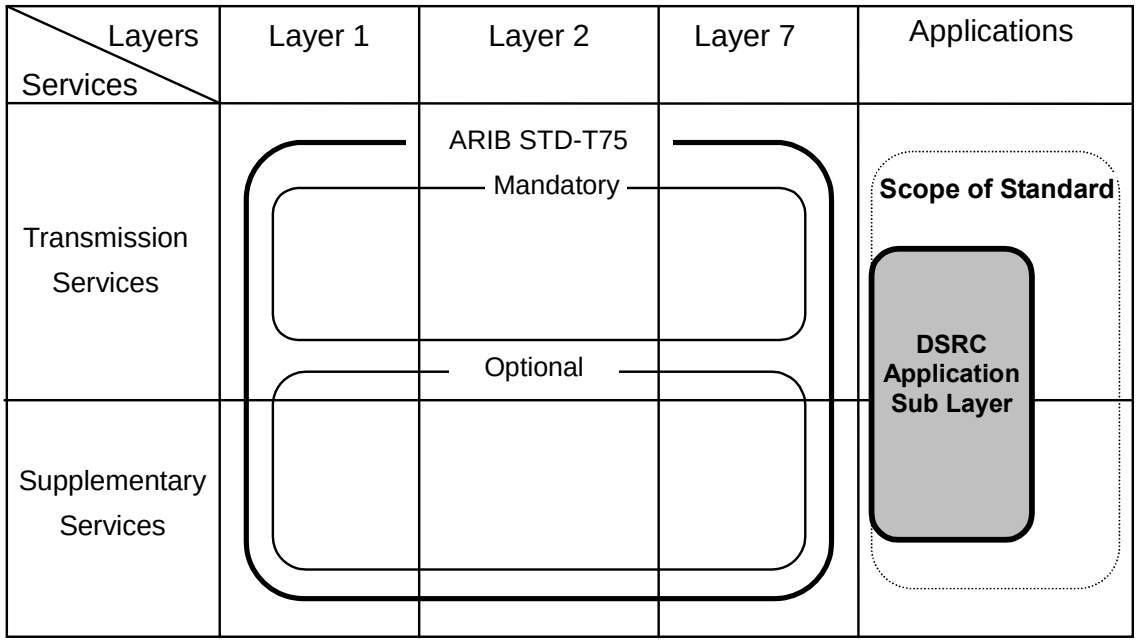


Figure 1.3-1 — Scope of standardization

1.4 References

This Standard incorporates provisions from other publications by dated or undated reference. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this Standard only when incorporated in it by amendment or revision. For undated references the latest edition of the publication referred to applies.

1.4.1 Normative References

ARIB STD-T75	DEDICATED SHORT-RANGE COMMUNICATION (DSRC) SYSTEM
ARIB TR-T16	DEDICATED SHORT-RANGE COMMUNICATION (DSRC) SYSTEM TEST ITEMS AND CONDITIONS FOR LAND MOBILE STATION COMPATIBILITY CONFIRMATION
ARIB TR-T17	TEST ITEMS AND CONDITIONS FOR DEDICATED SHORT-RANGE COMMUNICATION (DSRC) APPLICATION SUB LAYER LAND MOBILE STATION COMPATIBILITY CONFIRMATION
ISO/IEC 8824-1	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation
ISO/IEC 8825-2	Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)

1.4.2 Informative References

ANSI/IEEE std 802.3	CSMA/CD Access Method and Physical Layer Specifications
IETF RFC 1661	The Point-to-Point Protocol (PPP)
IETF RFC 1662	PPP in HDLC-like Framing
IETF RFC 3232	Assigned Numbers
IETF RFC 768	User Datagram Protocol

Empty page

2 System Overview

2.1 System Configuration

This system consists of a Base Station and a Land Mobile Station (or abbreviated "Mobile Station") as specified in the ARIB standard STD-T75.

2.2 System Basic Function

Following applications can be supported in the system. Refer to Annex C for application examples using the DSRC-ASL.

(1) Network Applications on Internet Protocol (IP)
(2) Network Applications on Non-Internet Protocol (IP). It could be supported. However, this application is out of the scope in this version.
(3) Non-Network Applications, that do not use the network protocol stacks as described in (1) (2) and can directly interface on the DSRC-ASL.

2.3 Structure of the Application Sub Layer Core

2.3.1 Protocol Stack

The DSRC-ASL interfaces between DSRC protocol stacks and Network applications or Non-Network applications, and it provides supplemental communication functions for DSRC communication functions. The structure of the core of the DSRC-ASL is shown in Figure 2.3-1. It provides a platform for DSRC applications without awareness of lower layer DSRC protocol stacks.

The DSRC-ASL as shown in Figure 2.3-1 consists of extended link control protocol (ASL-ELCP), which interfaces with DSRC protocol stacks and conducts fundamental application processes, and a network communication control protocol (ASL-NCP). ASL-NCP also consists of plural communication control protocols such as an LAN Control Protocol (LANCP), which can interface various types of network protocols.

The ASL-ELCP provides plural complementary communication control protocols such as, a client-server type communication control and/or a bulk transmission control. The ASL-NCP realizes the interface with plural network protocols and could treat various types of network specifications. The ASL-ELCP also provides the DSRC communication connection management functions for easy deployment of general DSRC applications.

Complementary functions provided by the ASL-ELCP are also selectable optional functions for easy applied to variety types of mobile stations, such as a low cost (implementing basic function) mobile station or a high-grade mobile station (implementing plural functions), to choice required functions in accordance with the subject DSRC application.

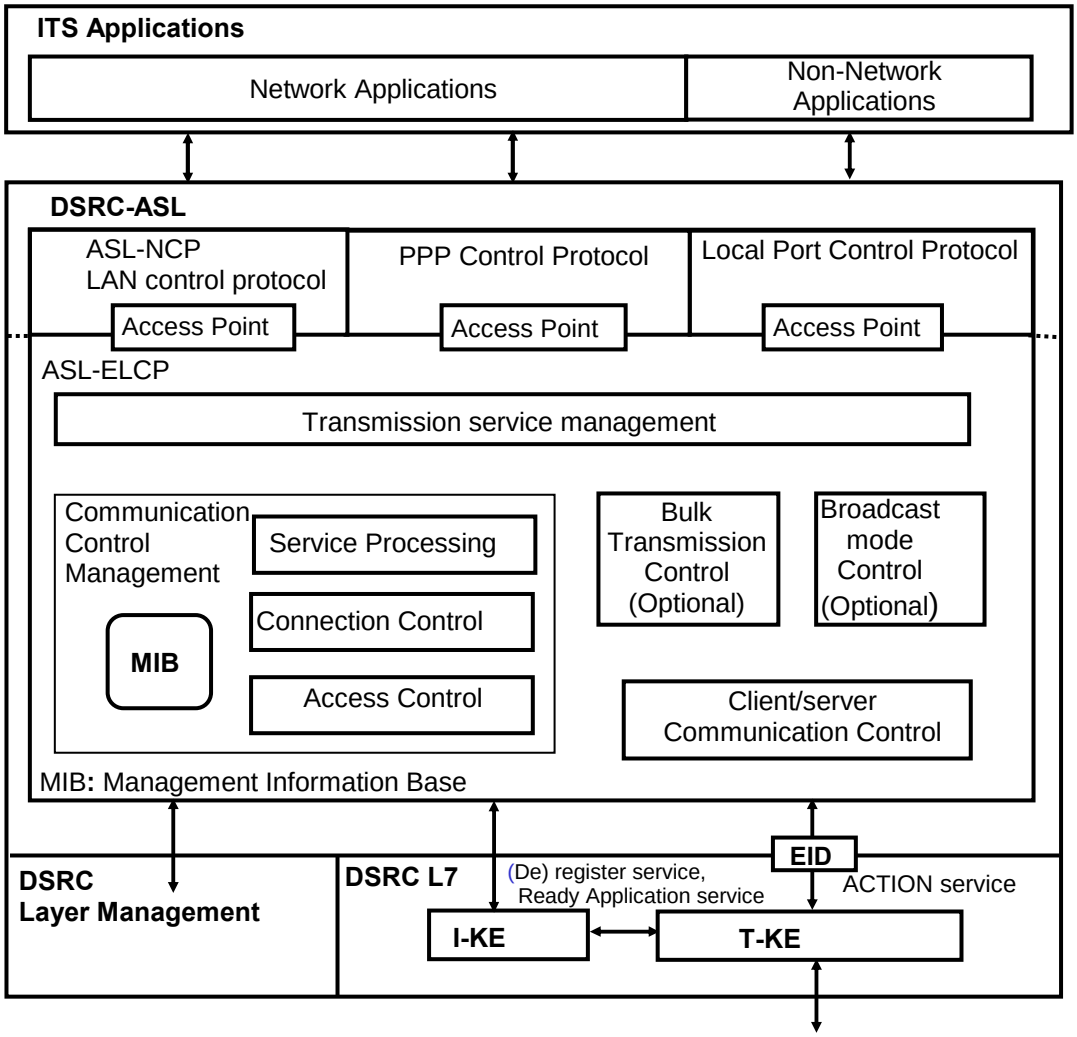


Figure 2.3-1 — Structure of the Application Sub Layer Core

2.3.1.1 Extended Link Control Protocols (ASL-ELCP)

The ASL-ELCP has functions for entities and each peer entity has a common protocol as follows.

– Services provision to the ASL-NCP
– Data Transmission service (process) by identifying destination address of receiving data unit
– Client-Server type communication control
– Bulk transmission control (Optional)
– Broadcast mode control (Optional)
– Communication control management

Communication control management has functions as follows.

– Communication connection control management function for maintaining connection

– Access control management function for the accesses to the base station

– Event report management function in occurred in the ASL-ELCP

– Registration management function for the registration of the ASL-ELCP MIB (Management Information Base)

2.3.1.2 Network Control Protocol (ASL-NCP)

The ASL-NCP conducts the capsulation of plural protocols, the establishment of access points and the setting the protocol type. The ASL-NCP also consists of plural control protocols for various types of connected network protocols.

The ASL-NCP consists of a Point-to-Point-Protocol Control Protocol (PPPCP) and an LAN Control Protocol (LANCP) for the connection to network application, and a Local Port Control Protocol (LPCP) for the connection to non-network application.

2.3.2 Procedure Outline

A procedure outline of DSRC-ASL is shown in Figure 2.3-2.
A DSRC Layer 7 establishes a DSRC communication link and an ASL-ELCP is activated by a notification of a DSRC communication link establishment from the DSRC Layer 7. After the activation, the ASL-ELCP, at first, compares its own ASL profile with a peer ASL profile passed through the established DSRC communication link and confirms the available functions in the ASL-ELCP. In this procedure, the ASL-ELCP does not conduct any setting related to the ASL-NCP.

After the confirmation of the ASL profiles, when an access management function is usable, a peer authentication is conducted. In case of the successful authentication, the ASL-ELCP activates each ASL-NCP and move state to ASL-NCP process phase.

The activated ASL-NCP conducts the initial setting for corresponding ASL-NCP. Each network protocol cannot be activated after the completion of the initial setting for corresponding ASL-NCP.

After completion of the above procedure, the state moves to a communication phase and starts a communication using the network protocol.
As described above, the network protocol, such as IP, is usable.

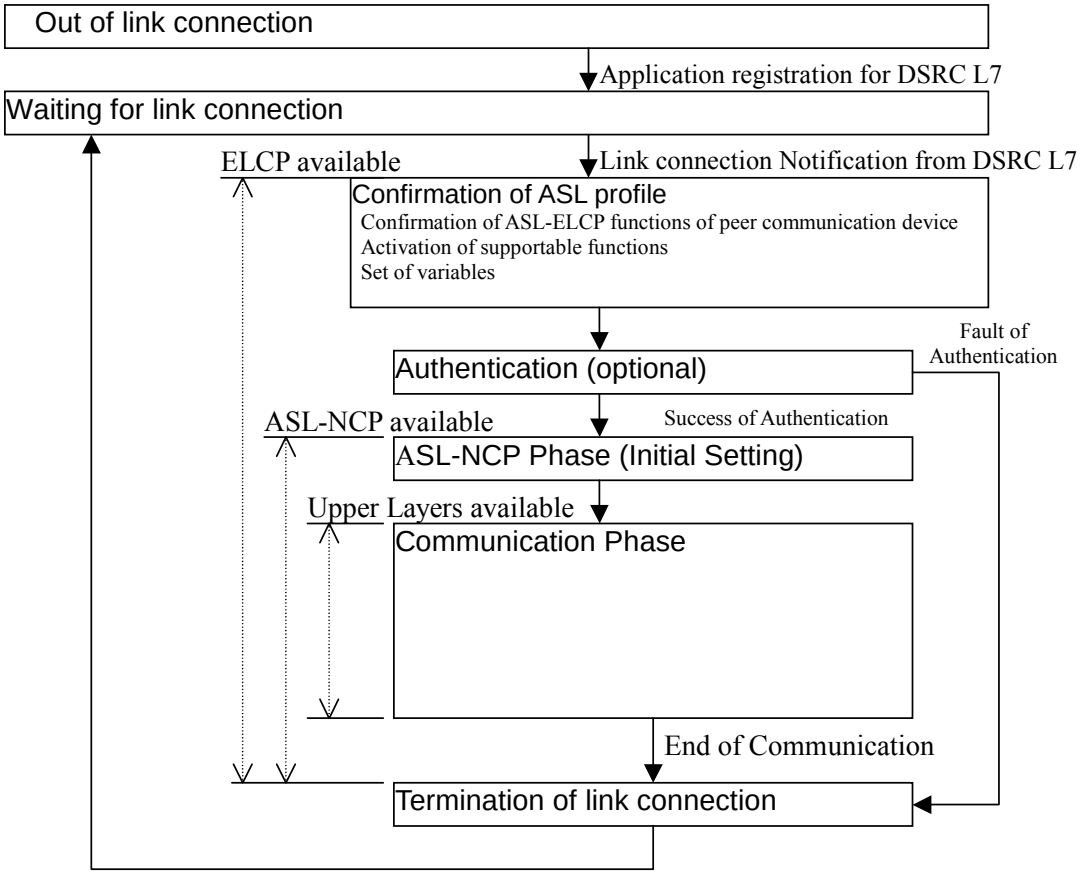


Figure 2.3-2 — Outline of the ASL-ELCP Operation Procedure

2.3.3 Identification of the Connection with a peer ASL-NCP

For sending a data unit to a correct peer application, as shown in Figure 2.3-3, the DSRC-ASL has access points to identify the ASL-NCP on the ASL-ELCP. Each application can identify the peer application connection through this access point identification.

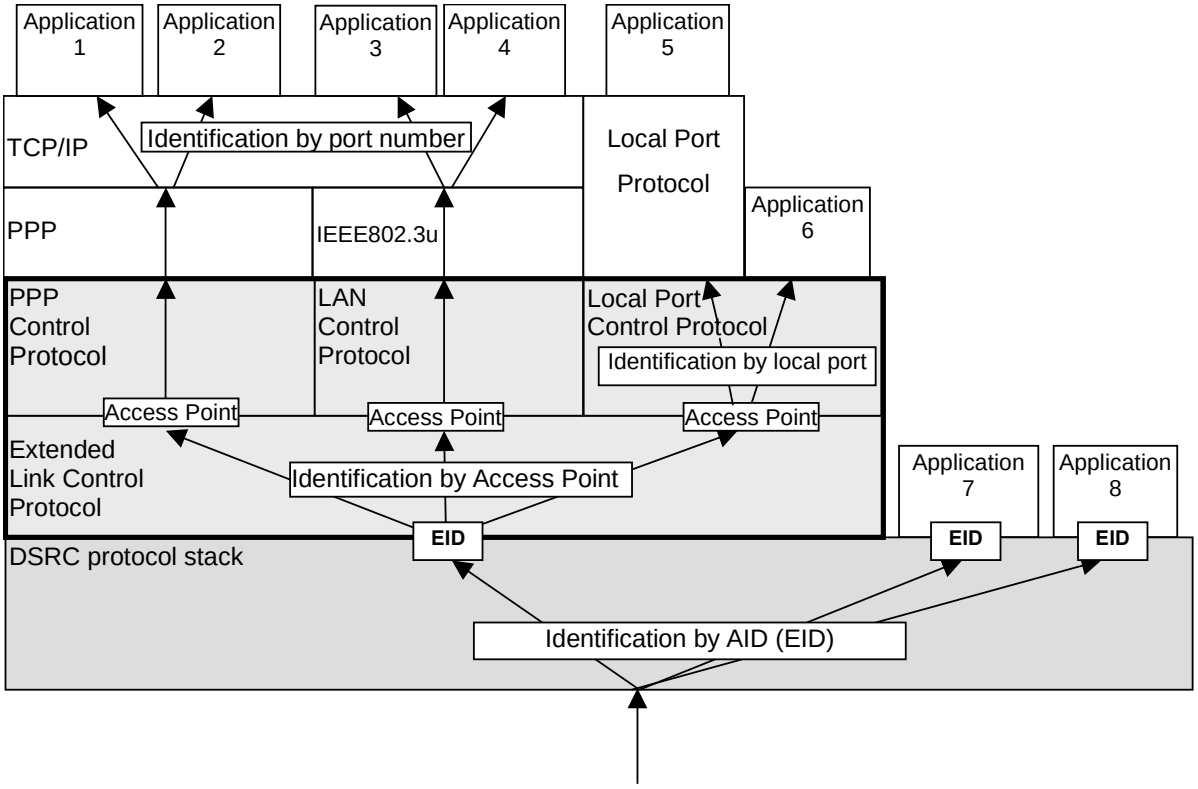


Figure 2.3-3 — Connection Identification Concept

2.3.3.1 Concept of Connection Identification for Network Applications

Figure 2.3-4 shows the concept of the access point identification on IP applications. In principal, a data unit passing from the upper layer to the lower layer, new identification numbers (Containing IP address、Application Identifier (AID), and Link Identifier (LID)) are added and transmitted to a peer entity. The peer receiving entity determines the transmission application in accordance with attached these identification numbers.

In actual transmission process, in accordance with transmission control information, such as a Transmission Control Protocol (TCP) Header and an IP Header, the port numbers and/or addresses of the destination application and the source application are indicated.

In the DSRC, the LID is not passed with the application data unit. Furthermore, when a base station establishes communication links with plural mobile stations, for example, the LANCP of the base station manages a pair of the LID and Media Access Control (MAC) address.

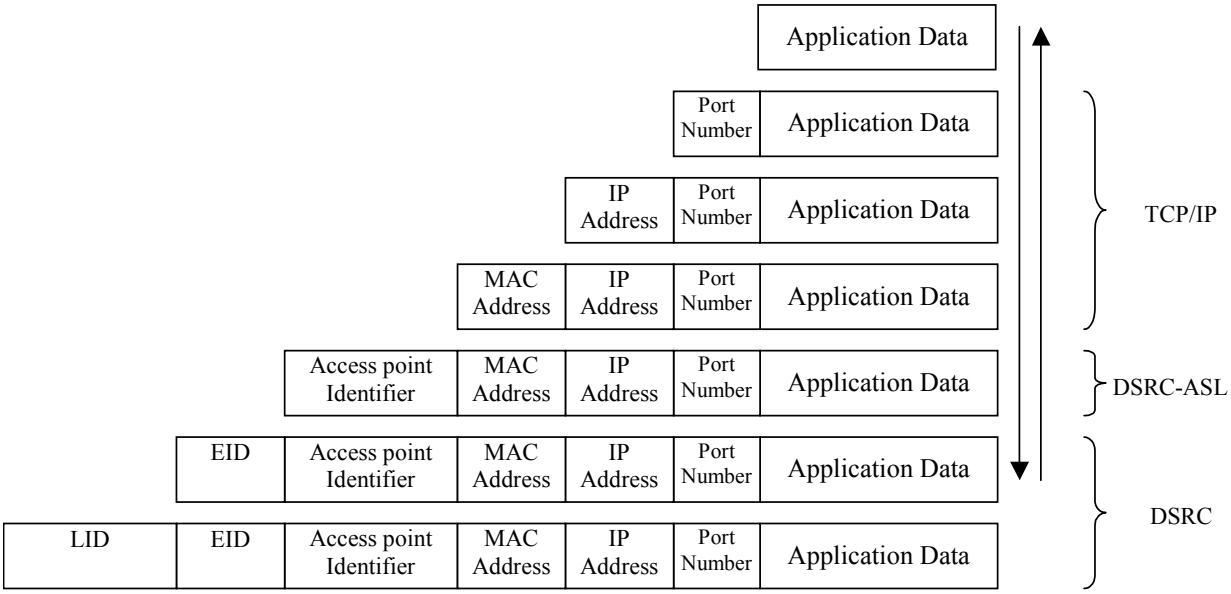


Figure 2.3-4 — Concept of Connection Identification for Network Applications

2.3.3.2 Concept of Connection Identification for Non-Network Applications

Figure 2.3-5 shows the concept of the access point identification on non-network applications. An application data unit passes to lower layer with a local port number and an access point identifier.

It is assumed that the local port number uses for the identification of the corresponding non-network application in case of existing plural non-network applications.

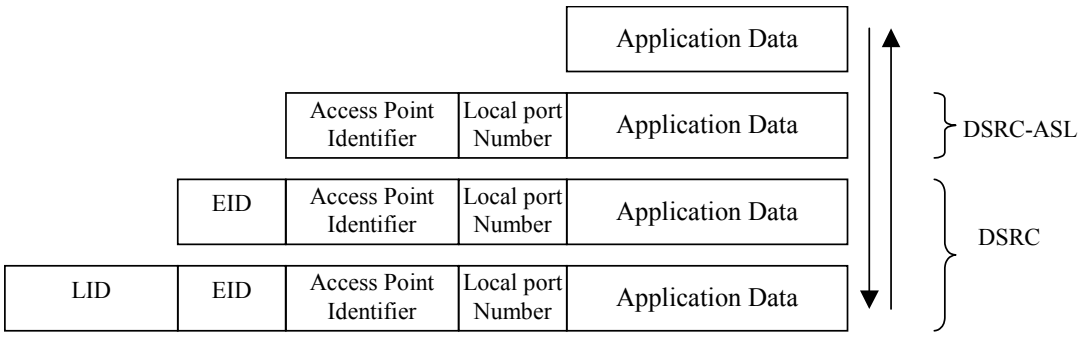


Figure 2.3-5 — Connection Identification Concept for Non-Network Applications

3 Communication Control

3.1 Overview

This clause specifies the communication control method of the DSRC-ASL. The interfaces are defined according to the protocol model shown in clause 2.

Figure 3.1-1 shows the overview of the service interfaces and the protocol stacks of the DSRC-ASL.

The ASL-ELCP exchanges the protocol data units (PDU) to the peer ASL-ELCP by using the service interface that DSRC layer 7 provides. The ASL-ELCP conducts the communication procedure provided for in the ASL-ELCP. And the ASL-ELCP provides the ASL-NCP the service interface of the communication service for the data transmission and the management service for the management control.

The ASL-NCP exchanges the protocol data units (PDU) between the peer ASL-NCP by using the service interface that the ASL-ELCP provides. The ASL-NCP conducts the communication procedure provided for in the ASL-NCP. The service interface, which the ASL-NCP offers to the upper layer protocol, provides only for mutual operation with the upper layer protocol, and doesn't provide for the service specification as a rule.

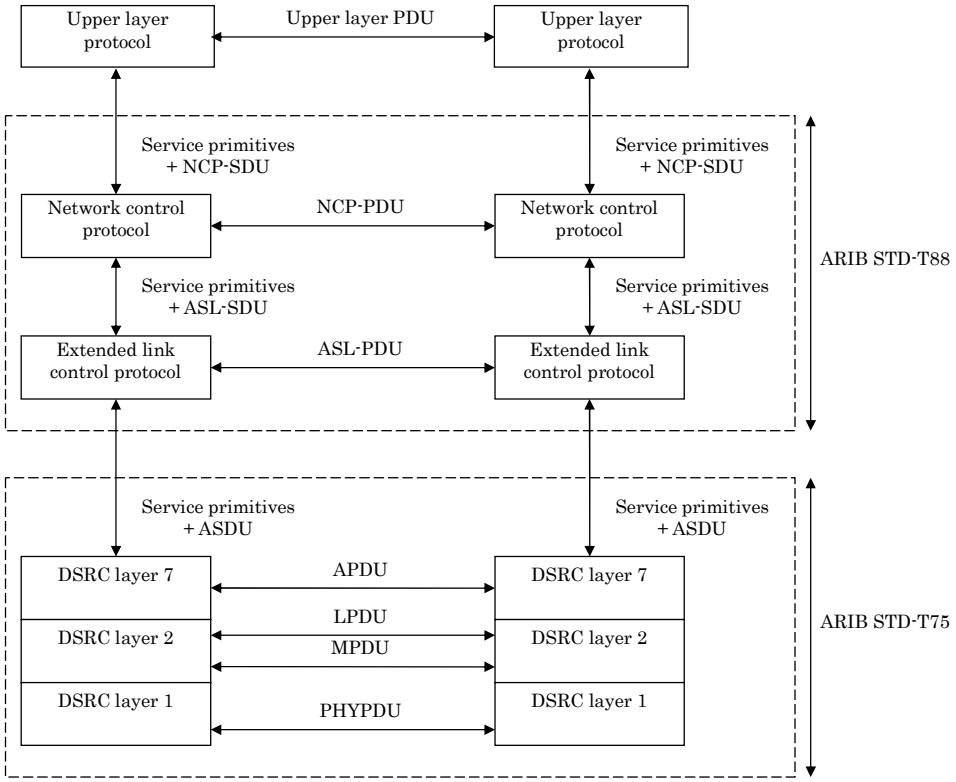


Figure 3.1-1 — Overview of the Service Interfaces and the Protocols of the DSRC-ASL

Empty page

3.2 Extended link Control Protocol (ASL-ELCP)

3.2.1 Overview

3.2.1.1 Function

The ASL-ELCP has the following functions, in order to complement the communication facility of the DSRC, and it provides the communication services for data transmission and the management service to control the ASL-NCP.

(1) Multi-protocol correspondence
(2) Client / Server type communication control
(3) Bulk transmission control
(4) Broadcast transmission mode control
(5) Access control
(6) Communication connection management

3.2.1.2 Structure of the ASL-ELCP

Figure 3.2-1 shows the ASL-ELCP structure of a base station, and Figure 3.2-2 shows the ASL-ELCP structure of a mobile station. The ASL-ELCP is composed of the entity of each functional processing module in order that the function may be added or deleted.

The communication control management entity, which manages the ASL-ELCP, assigns access-point identifiers of the same kind of the ASL-NCP in order to connect the peer protocol between management entities. And the communication control management transmits and receives the data between management entities using the communication service interface of extended communication control.

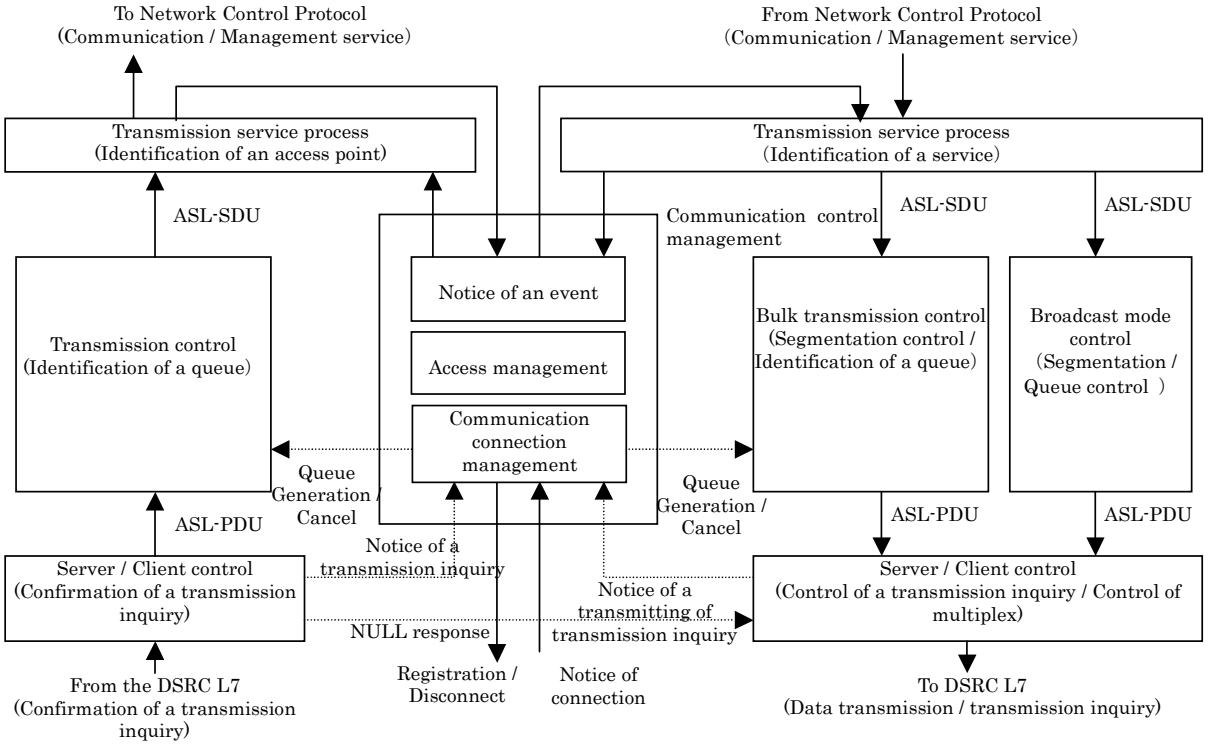


Figure 3.2-1 — ASL-ELCP Structure of the Base Station

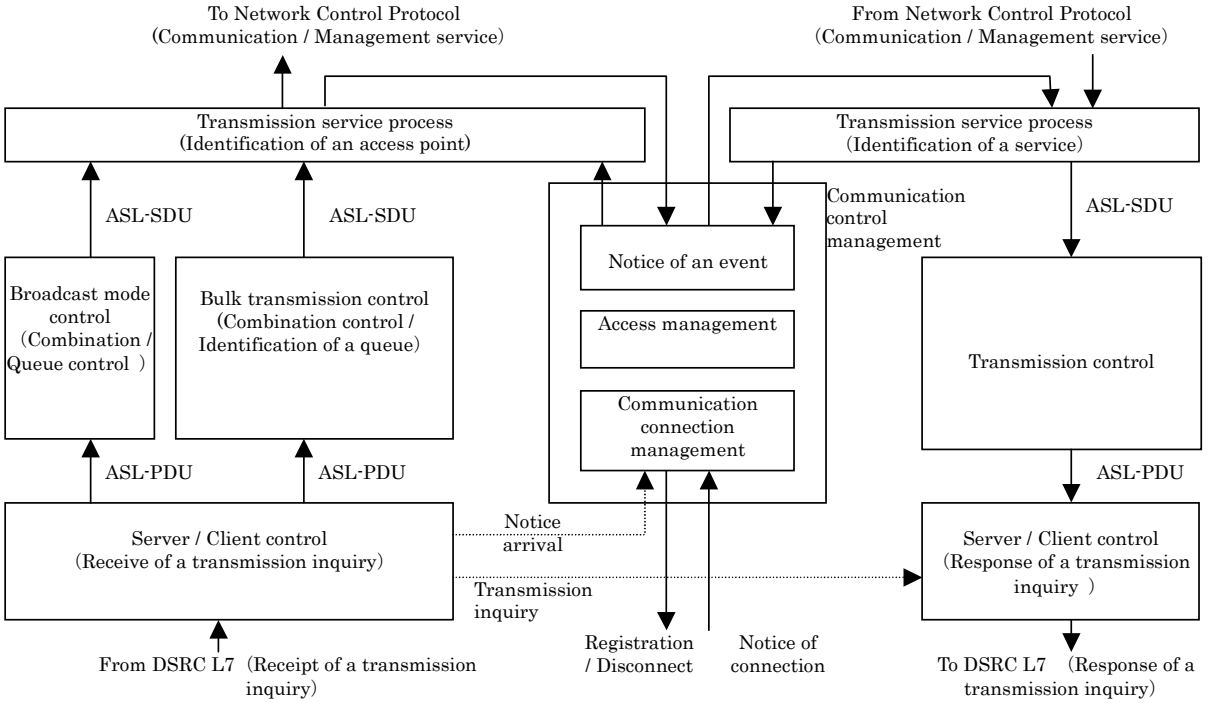


Figure 3.2-2 — ASL-ELCP Structure of the Mobile Station

3.2.1.3 Service Interface

The transmission service entity is allocated to interface between this ASL-ELCP and the ASL-NCP, and the ASL-ELCP provides the ASL-NCP with the communication service for data transmission and the management service for management control.

And the client / server type communication control entity is allocated to interface between this ASL-ELCP and the DSRC layer 7, and the ASL-ELCP controls the communication using the ACTION Primitive which is provided by the DSRC layer 7.

3.2.1.4 Protocol

The communication control procedure of ASL-ELCP is defined by the communication control information, which is added by the ASL-ELCP. This communication control information is added to the communication control service data unit (ASL-SDU: ASL Service Data Unit) received from the ASL-NCP as shown in Figure 3.2-3.

And communication control management procedure is defined by the access control information, which is added by the ASL-NCP.

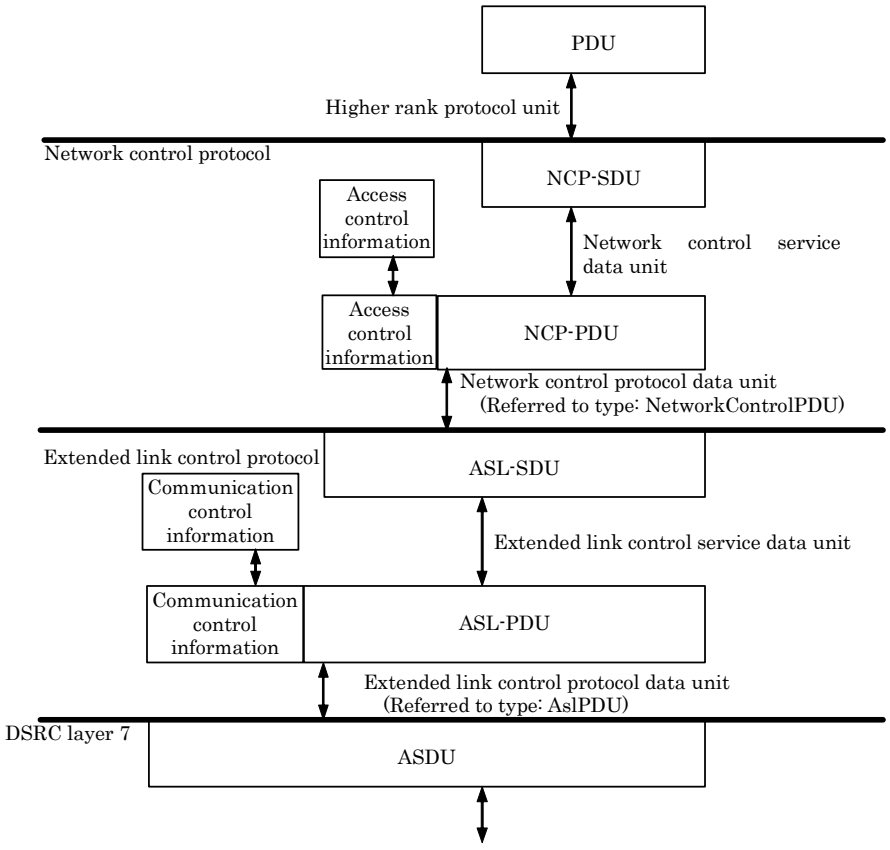


Figure 3.2-3 — Relationship between Data Structure in each Protocol Layers

3.2.2 Extended Link Control

3.2.2.1 Communication Service Interface

3.2.2.1.1 Overview of Primitive Relationship

The Communication control of the ASL-ELCP provides the ASL-NCP with the following primitives as communication service.

SendDataUnit.request
SendDataUnit.indication

The SendDataUnit.request is sent to the ASL-ELCP from the ASL-NCP, in order to request that the ASL-SDU passed from the ASL-NCP is transmitted to the remote station. The SendDataUnit.indication is sent to the ASL-NCP from the ASL-ELCP, in order to show the arrival of the ASL-SDU.

3.2.2.1.2 Service Contents Specification

In this sub-clause, primitives and parameters about the communication service are specified. Parameters as an interface are described abstractly. And the information, which is needed for a receiving peer entity, is specified. However, the concrete realization methods, which provide with this information, are not specified.

The parameter “linkAddress” discriminates the access points of remote stations for each ASL-NCP. An ASL-SDU may be actually stored in the parameter “parameter” or a pointer of an ASL-SDU may be stored in it. Moreover, an ASL-SDU may also pass by other methods.

In the ASL-ELCP, these parameters passed from the ASL-NCP are defined as the following specifications in Table 3.2-1.

Table 3.2-1 — Parameters of the Communication Service

Parameter name	ASN.1 type	Remarks
LinkAddress	DsrcLID	
Parameter	NetwokControlPDU	

The logical relationship between communication service primitives, which are provided by the ASL-ELCP to the ASL-NCP, is shown in Figure 3.2-3.

NOTE Service primitive type abbreviation is as follows: “request” is abbreviated to “req” and “indication” is abbreviated to “ind”. This abbreviation is used through this standard.

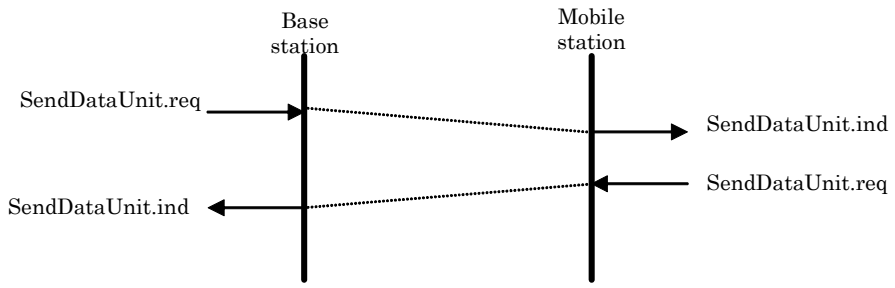


Figure 3.2-4 — Logical Relationship between Communication Service Primitives

3.2.2.1.2.1 Data Transfer Request Primitive

(1) Function
This primitive is service primitive which request that the ASL-SDU is transmitted to remote station.

(2) When Generated
The ASL-NCP always generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

SendDataUnit.request (linkAddress, parameter)

The parameter “linkAddress” stores the link address, which is used in the DSRC. In the mobile station, the private link address is specified. In the base station, the private link address or the multicast link address is specified. In addition, when the multicast link address is specified, the ASL-SDU is delivered as the broadcast mode.

The parameter “parameter” stores the ASL-SDU passed from the ASL-NCP in the transmitting station.

3.2.2.1.2.2 Data Arrival Notify Primitive

(1) Function
This primitive is a service primitive, which notifies of the arrival of the ASL-SDU from the remote station.

(2) When Generated
When the arrival of the ASL-SDU is shown, the ASL-ELCP generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

SendDataUnit.indication (linkAddress, parameter)

The parameter “linkAddress” stores the link address, which is used in the DSRC. In the mobile station, the private link address is specified. In the base station, the private link address or the multicast link address is specified.

The parameter “parameter” stores the arrival of the ASL-SDU.

3.2.2.2 Protocol Data Unit (PDU)

3.2.2.2.1 PDU Format

The Protocol data unit of communication control (ASL-PDU: ASL Protocol Data Unit) shown in is the DSRC-ASL PDU. And this PDU consists of the ASL-SDU, which is passed from the ASL-NCP, and the control field (communication control information), which specifies the control information for directing the procedure of the ASL-ELCP.

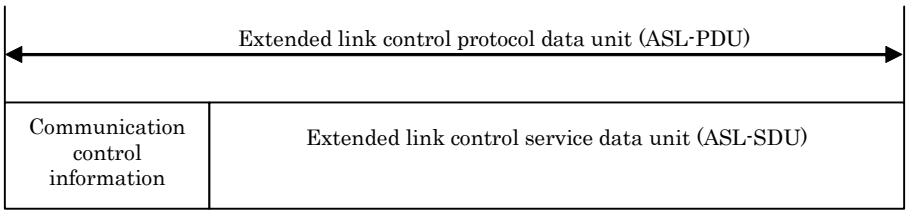


Figure 3.2-5 — PDU Format between the DSRC and the ASL

3.2.2.2.2 PDU Element

3.2.2.2.2.1 Connection Identification

The link address for identification of connection, the element identifier EID (Element Identifier) and etc. receive as a parameter of the service primitive, which the DSRC layer 7 provides.

3.2.2.2.2.2 Format of the Control Field

The communication control information specifies the control field for directing the procedure of the ASL-ELCP. This content provide in sub-clause 3.2.2.3.1.

3.2.2.2.2.3 Format of the Information Field

The ASL-PDU passed from the ASL-NCP is divided into bulk segments or penetrated, and stored in the information field.

3.2.2.3 Procedure Elements of the Extended Communication Control

3.2.2.3.1 Communication Control Information Format

The communication control information specifies the control field for directing the procedure of the ASL-ELCP, and it is held in common between the base station and the mobile station in order to perform procedure.

The format of this communication control information field is shown in Table 3.2-2. In addition, this communication control information is defined by the parameter "aslLinkProtocol" of the "AslPDU" type, which define the ASL-PDU format of the ASL-ELCP.

Table 3.2-2 — Communication Control Information (aslLinkProtocol) Field Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Preamble	bulk Enable	bulkTermination	pduGroup				
2	SegmentNumber							
3	broadcastParameter（optional）							
		dummy				serviceTime		
4		serviceTime						

(1) Bulk transmission valid identifier (bulkEnable)
This identifier indicates that bulk transmission is valid or invalid.

When an ASL-PDU is bulk segment, which is bulk transferred, this identifier of the ASL-PDU shall be true.

When the ASL-PDU shall not process segmentation, this identifier is false.

(2) Valid identifier of the bulk transmission (bulkTermination)
This identifier indicates the last segment of bulk transmission.

When an ASL-PDU is the last segment, which is bulk transferred, this identifier of the ASL-PDU shall be true.

When the ASL-PDU shall not process segmentation, this identifier is false.

(3) PDU group number (pduGroup)
This number is the identification information of bulk segment.

A PDU group number is assigned to the ASL-SDU before division processing for the bulk transmission. The number shall be assigned by modulo "32" for each sending queue. And a

number shall be also assigned to the ASL-SDU, which is not adapted in the bulk transmission.

When the received bulk segment assembles the ASL-SDU, the segment of the same PDU group number shall be used.

(4) Segment number (segmentNumber)
This number is a serial number, which shows the division order of a bulk segment.

The segment number of first divided bulk segment shall be set to "0", and the serial number shall be set to the value which is incremented in order. When the bulk segment is assembled, the order of the segment number is guaranteed. When a bulk transmission is not applied, the segment number shall be set to "0".

(5) Auxiliary parameter for the broadcast mode (broadcastParameter)
The Auxiliary parameter for the broadcast mode is an option. When the ASL-SDU is segmented on the broadcast mode processing, the following parameter is added to the communication control information of the bulk segment.

(a) Value of connection guard timer (serviceTime)
This value shall be set to the connection timer of the mobile station (CTO: Connection Timer for OBU) in applicable communication area. When the point-to-point communication is not performed (does not process association procedure), the mobile station manages the validity time of broadcast service using this value.

A unit shall be millisecond, and the value shall be set to range of "0" to "4095". In addition, when it shows that the time is infinite, the value shall be set to "0".

NOTE When the point-to-point communication is performed in parallel with broadcast communication, the mobile station uses the timer value given by the ASL base station profile.

3.2.2.3.2 Communication Control Parameters

3.2.2.3.2.1 Maximum Receiving Data Unit Length of the ASL-ELCP (MRU)

The MRU is the maximum receiving length of the data when the ASL-ELCP receives from the ASL-NCP. The MRU of the ASL-ELCP shall be the larger value in the maximum transmission unit (MTU: Maximum Transmission Unit), which is specified by the implemented ASL-NCP.

3.2.2.3.2.2 Segment Unit for the Unicast of Bulk Transmission (SUU)

The SUU is a unit, which segments the ASL-SDU, when the bulk transmission is performed. This unit shall be set to the following value according to the selected profile.

The value of SUU shall be set to 56 octets in case of the ASK (Amplitude Shift Keying) system, and set to 183 octets in case of the π/4 shift QPSK (Quadrature Phase Shift Keying) system.

3.2.2.3.2.3 Segment Unit of the Broadcast Mode Control (SUM)

The SUM is a unit, which segments the ASL-SDU, when the bulk transmission is performed on broadcast mode control. This unit is set to the following value according to the selected profile.

The value of the SUM is set to 54 octets in case of the ASK system, and set to 181 octets in case of the π/4 shift QPSK system.

3.2.2.3.2.4 Repetition Transmitting Number of the Broadcast Mode Control (k)

The k is a number of times, which the ASL-PDU sends repeatedly on the broadcast mode control. Since the receiving error rate of the data, which is sent by broadcast, is different value according to the value of k, it needs to be decided in consideration of the reliability of a system. As reference, the relation between a receiving error rate and k is shown in Annex F.

3.2.2.3.3 Communication Services from DSRC Layer7 Interface

The specifications of the interface of communication service, which is provided for by the DSRC layer7 specified in the ARIB STD-T75, are provided for in this sub-clause. The ASL-ELCP performs the client / server type communication control in order to send or receive the ASL-SDU which is generated by the ASL-NCP. At this time, the ASL-ELCP uses the ACTION primitive provided by the DSRC layer 7, which is able to define the two or more functions.

The specification of the ACTION primitives specified in the ARIB STD-T75 is as follows:

ACTION.request ([iid], lid, chaining, eid, actionType, [accessCredentials],
[actionParameter], mode, flowControl)
ACTION.indication ([iid], lid, chaining, eid, actionType, [accessCredentials],
[actionParameter], mode, flowControl)
ACTION.response ([iid], lid, chaining, eid, flowControl,
[responseParameter], [ret])
ACTION.confirm ([iid], lid, chaining, eid, [flowControl],
[responseParameter], [ret])

3.2.2.3.3.1 Overview of the Sub-primitive Relationship

The ASL-ELCP defines and uses the following services and sub-primitives, using the

ACTION primitives provided by DSRC layer 7.

(1) Data transfer service
In the data transfer service; the following sub-primitives are defined using the ACTION primitives.

SendMessage.request
SendMessage.indication

The “SendMessage.request” is sent to the DSRC layer 7 of the base station from the ASL-ELCP of the base station, in order to request that the ASL-PDU generated by the ASL-ELCP of the base station is transmitted to the mobile station.

The “SendMessage.indication” is sent to the ASL-ELCP of the mobile station from the DSRC layer 7 of the mobile station, in order to notify the ASL-ELCP of mobile station that the ASL-PDU arrived.

(2) Data transfer inquiry service
In the data transfer inquiry service; the following sub-primitives are defined using the ACTION primitives.

WaitMessage.request
WaitMessage.indication
WaitMessage.response
WaitMessage.confirm

The “WaitMessage.request” is sent to the DSRC layer 7 of the base station from the ASL-ELCP of the base station, in order to confirm the existence of the ASL-PDU, which transmits to a base station from a mobile station. The “WaitMessage.indication” is sent to the ASL-ELCP of the mobile station from the DSRC layer 7 of the mobile station, in order to notify the ASL-ELCP of the mobile station that the transmission inquiry arrived. The “WaitMessage.response” is sent to the DSRC layer 7 of the mobile station from the ASL-ELCP of the mobile station, in order to send the response of the transmission inquiry. The “WaitMessage.confirm” is sent to the ASL-ELCP of the base station from the DSRC layer 7 of the base station, in order to notify the ASL-ELCP of the base station that the response of the transmission inquiry arrived.

3.2.2.3.3.2 Service Contents Specification

In this sub-clause, the primitive and parameter about the communication service are specified. The parameter as an interface is described abstractly and the information, which is needed for a reception side entity is specified. However, the concrete realization method, which provide with this information is not specified.

The parameter “linkAddress” identifies the access points between the remote stations of every ASL-NCP.

An ASL-SDU is actually stored in the parameter “dataPcket” or a pointer of the ASL-SDU is stored in it. Moreover, ASL-SDU may also pass by other methods.

In the ASL-ELCP, these parameters are dealt with by the following specifications in Table 3.2-3.

Table3.2-3 — Parameters of the Data Transfer Request Primitive

Parameter name	ASN.1 type	Remarks
linkAddress	DsrcLID	
dataPacket	AslPDU	

The logical relationships between sub-primitives, which are defined by the ACTION primitives, are shown in Figure 3.2-6.
NOTE Service primitive type abbreviation is as follows: “response” is abbreviated to “res” and “confirmation” is abbreviated to “conf”. This abbreviation is used through this standard.

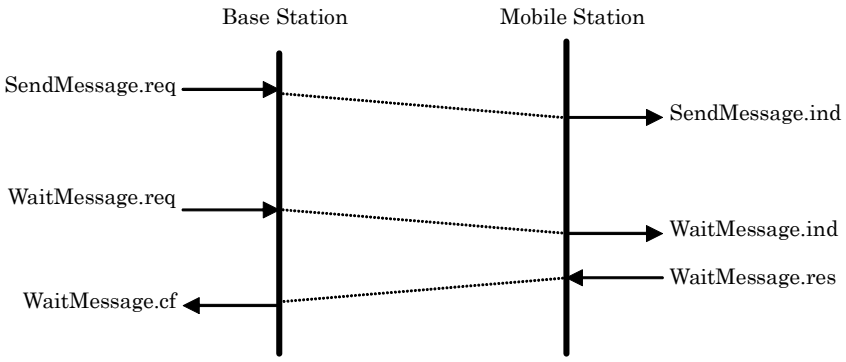


Figure 3.2-6 — Logical Relationship between Sub-primitives defined by the ACTION Primitives

3.2.2.3.3.3 Data Transfer Services

The base station uses the data transfer service in order to send the data to a mobile station from a base station. There is no response in this service. The data sent from a mobile station is gained with the data transmission inquiry service shown in sub-clause 3.2.2.3.3.4.

3.2.2.3.3.3.1 Data Transfer Request Primitive

(1) Function
This sub-primitive is service primitive which requests that the ASL-SDU is sent to remote station from base station.

(2) When generated

The ASL-NCP of base station always generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

SendMessage.request (linkAddress, dataPacket)

The parameter “LinkAddress” stores the private link address, which is specified by the ASL-ELCP or the multicast link address, which is set to “0” or “1”.

The parameter “dataPacket” stores the ASL-PDU, which is generated by the ASL-ELCP of the base station.

3.2.2.3.3.3.2 Data Transfer Notify Primitive

(1) Function
This primitive is service primitive, which notifies the arrival of the ASL-PDU from the base station.

(2) When generated
When the arrival of the ASL-PDU is shown, the ASL-ELCP of the mobile station generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

SendMessage.indication (linkAddress.dataPacket)

The parameter “linkAddress” stores the private link address or the multicast link address, which is set to “0” or “1”.

The parameter “dataPacket” stores the PDU, which is generated by the ASL-ELCP of the base station.

3.2.2.3.3.4 Data Transfer Inquiry Services

The data transfer inquiry service is that the base station awaits the transmission data from a mobile station.

The base station inquiries a mobile station by the request primitive of the data transmission inquiry service, and the mobile station responses by the response primitive of the data transmission inquiry service.

3.2.2.3.3.4.1 Data Transfer Inquiry Request Primitive

(1) Function
This sub-primitive is a service primitive, which requests that the ASL-ELCP of the base station inquire a data transmission to a mobile station.

(2) When generated
The ASL-ELCP of the base station generates this sub-primitive according to the schedule of transmission, which the ASL-ELCP of the base station sets up.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

WaitMessage.request (linkAddress, [dataPacket])

The parameter “linkAdress” shall be the private link address.

The parameter “dataPacket” shall store the ASL-PDU when the transmission data from the base station exists. When the transmission data does not exist, this parameter “dataPacket” shall be “omitted”.

3.2.2.3.3.4.2 Data Transfer Inquiry Notify Primitive

(1) Function
This sub-primitive is a service primitive, which notifies of the arrival of the data transfer inquiry from the base station.

(2) When generated
When the arrival of the data transmission inquiry from the base station is shown, the ASL-ELCP of the mobile station generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

WaitMessage.indication (linkAddress, [dataPacket])

The parameter “linkAddresss” shall be the private link address. The parameter “dataPacket” stores the PDU when the transmission data from base station exits. When the transmission data does not exist, this parameter “dataPacket” shall be “omitted”.

3.2.2.3.3.4.3 Data Transfer Inquiry Response Primitive

(1) Function
This sub-primitive is a service primitive, which requests the response to the data transfer inquiry from a base station.

(2) When generated
When the arrival of the data transfer inquiry from the base station is shown, the ASL-ELCP of the mobile station generates this sub-primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

WaitMessage.response (linkAddress, [dataPacket])

The parameter “linkAddresss” stores the private link address.

The parameter “dataPacket” shall store the PDU when the transmission data from the base station exists. When the transmission data does not exist, this “dataPacket” shall be “omitted”.

3.2.2.3.3.4.4 Data Transfer Inquiry Confirmation Primitive

(1) Function
This sub-primitive is a service primitive, which notifies of the arrival of the response to the data transfer inquiry.

(2) When generated
When the arrival of the data transfer inquiry is shown, the ASL-ELCP of the base station generates this sub-primitive.

(3) Semantics of the service primitive
This primitive shall provide parameters as follows.

WaitMessage.confirm (linkAddress, [dataPacket])

The parameter “linkAddress” specifies the private link address.

The parameter “dataPacket” shall store the PDU when the transmission data from base station exists. When the transmission data does not exist, this parameter “dataPacket” shall be “omitted”.

3.2.2.3.3.5 ACTION Primitive Parameter Setting

The specifications of the ACTION primitives for the sub-primitives are shown below:

(1) Data transmission service
When the “SendMessage” sub-primitive is used, the parameters of the “ACTION.resuest” primitive shall be set to below (Refer to Table 3.2-4.).

Table 3.2-4 — ACTION.request Primitive Parameter Setting in the Data Transfer Service

Parameter name	ASN.1 type	Value	Remarks
Mode	BOOLEAN	0	No response
Eid	Dsrc-EID	3 or any value from 4 to 127	When the link address is the multicast link address, the “3” is used.
actionType	ActionType	119	
accessCredentials	OCTETSTRING (SIZE(0..127,..) OPTIONAL	-	Do not use
actionParameter	Container	octetstring: encodedDataPacket	The encodedDataPacket is a value with name, which shows the coding result by the type definition for the parameter dataPacket of the sub-primitive.
Iid	Dsrc-EID OPTIONAL	—	Do not use
Lid	DsrcLID	dsrcLinkAddress	The dsrcLinkAddress is a value with name, which is shown by the parameter linkAddress of the sub-primitive.
Chaining	-	-	Do not use
flowControl	-	1	DL-UNITDATA.request without response request

(2) Data transmission inquiry service

When the “WaitMessage” sub-primitive is used, the parameters of the “ACTION.request” and “ACTION.respons” shall be set to below (Refer to Table 3.2-5 or Table 3.2-6 respective.).

Table 3.2-5 — ACTION.request Primitive Parameter Setting in the Data Transfer Inquiry Service

Parameter name	ASN.1 type	Value	Remarks
mode	BOOLEAN	1	Response exist
Eid	Dsrc-EID	Any value from 4 to 127.	
actionType	ActionType	120	
accessCredentials	OCTETSTRING (SIZE(0..127,..) OPTIONAL	-	Do not use
actionParameter	Container	octetstring: “H/ encodedDataPacket	The encodedDataPacket is a value with name, which shows the coding result by the type definition for the parameter dataPacket of the sub-primitive.
Iid	Dsrc-EID OPTIONAL	—	Do not use
Lid	DsrcLID	dsrcLinkAddress	The dsrcLinkAddress is a value with name, which is shown by the parameter linkAddress of the sub-primitive.
Chaining	-	-	Do not use
flowControl	-	2	DL-UNITDATA.request with response request

Table 3.2-6 — ACTION.response Primitive Parameter Setting in the Data Transfer Inquiry Service

Parameter name	ASN.1 type	Value	Remarks
Eid	Dsrc-EID	Any value from 4 to 127.	
Iid	Dsrc-EID OPTIONAL	-	Do not use
responseParameter	Container OPTIONAL	octetstring: “H/ encodedDataPacket	The encodedDataPacket is a value with name, which shows the coding result by the type definition for the parameter dataPacket of the sub-primitive.
Ret	ReturnStatus OPTIONAL	status	The status is the value with name, which shows the process code according to the processing result. The process code is shown in the ARIB STD-T75.
Lid	Dsrc-LID	dsrcLinkAddress	The dsrcLinkAddress is a value with name, which is shown by the parameter linkAddress of the sub-primitive.
Chaining	-	-	Do not use
flowControl	-	1	DL-UNITDATA.request without response request

3.2.2.4 Extended Link Control Procedure

3.2.2.4.1 Data Transfer Service Process Procedure

3.2.2.4.1.1 Base Station Data Transfer Service

(1) Process of the data transfer service
When the request primitive of data transmitting (SendDataUnit.request) is called from the ASL-NCP, the ASL-ELCP shall get the ASL-SDU from the parameter “parameter”.

When the content of the parameter “linkAddress” is the private link address, the ASL-ELCP applies the bulk mode control procedure of the base station to the acquired ASL-SDU. And this procedure is specified by sub-clause 3.2.2.4.3.

However, in the following case, the ASL-SDU presupposes that it is invalid and shall not process.

(a) When the size of the ASL-SDU, which is passed by the parameter “parameter” exceeds the MRU of the ASL-ELCP, the request primitive is discarded. And the state “the size of data exceeded the maximum of a buffer” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP, which required the transmission.

(b) When the transmission queue is full, the ASL-SDU is discarded. And the state “the transmission queue is full, the request of transmission is failed” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP which required the transmission.

(c) When the parameter “linkAddress” is the multicast link address and that value of address is not “0”, the request primitive is canceled. And the state “the multicast link address is invalid.” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP which required the transmission.

(d) When the request primitive of data transmitting of which the parameter “linkAddress” is the multicast link address is passed to the ASL-ELCP which does not support the broadcast mode control transmission, the request primitive is canceled. And the state “this function is not supported.” is notified in the notice event primitive (EventInfomation.indication) to the ASL-NCP which required the transmission.

(2) Process of the receiving service
When the ASL-PDU is stored in the receiving queue, the ASL-SDU is distributed to the ASL-NCP by the notice primitive of data arrival (SendDataUnit.indication).

On this occasion, the communication control information in the ASL-PDU shall be deleted, and the ASL-SDU, which is passed to the ASL-NCP, shall be extracted from the ASL-PDU and be stored in the parameter “parameter”.

The link address of the ASL-PDU shall be stored in the parameter “parameter”.

And the ASL-NCP, which is the notice place of the notice primitive of data arrival (SendDataUnit.indication), shall be identified according to the access point identifier of the access control information field in the ASL-SDU.

On this occasion, when there is not the ASL-NCP, which is the notice place, the ASL-SDU shall be discarded.

3.2.2.4.1.2 Mobile Station Data Transfer Service

(1) Process of the transmission service
When the request primitive of data transmitting (SendDataUnit.request) is called from the ASL-NCP, the ASL-ELCP gets the ASL-SDU from the parameter “parameter”.

When the content of the parameter “linkAddress” is the private link address, the ASL-ELCP shall add the communication control information of the invalid bulk transmission mode to the acquired ASL-SDU and generate the ASL-PDU according to the rules in sub-clause 3.2.2.3.1, and the ASL-ELCP set it to the transmission queue.

However, in the following case, the ASL-SDU presupposes that it is invalid and shall not process.

(a) When the size of the ASL-SDU, which is passed by the parameter “parameter” exceeds the MRU of the ASL-ELCP, the request primitive is canceled. And the state “the size of data exceeded the maximum of a buffer” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP, which required the transmission.

(b) When the transmission queue is full, the ASL-SDU is discarded. And the state “the transmission queue is full, the request of transmission is failed” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP which required the transmission.

(c) When the parameter linkAddress is the multicast link address and that value of address is not “0”, the request primitive is canceled. And the state “the multicast link address is invalid.” is notified by the notice event primitive (EventInfomation.indication) to the ASL-NCP, which required the transmission.

(2) Process of the receiving service
When the ASL-SDU is obtained from the result of the bulk transmitting processing of the mobile station shown in sub-clause 3.2.2.4.2.2 or the broadcast control mode processing of the mobile station shown in sub-clause 3.2.2.4.3.4, the ASL-SDU is distributed to the ASL-NCP using the notice primitive of the data arrival.

And the ASL-SDU, which is passed to the ASL-NCP, shall be stored in the parameter “parameter”.

The link address of the ASL-SDU shall be stored in the parameter “linkAddress”.

And the ASL-NCP, which is the notice place of the notice primitive of data arrival (SendDataUnit.indication), shall be identified according to the access point identifier of the access control information field in the ASL-SDU.

On this occasion, when there is not the ASL-NCP, which is the notice place, the ASL-SDU shall be discarded.

3.2.2.4.2 Bulk Transmission Control Procedure

3.2.2.4.2.1 Bulk Transmission Process of the Base Station

The application propriety of the bulk transmission is decided according to the ASL mobile station profile, which is referred by the link address of the ASL-SDU of a processing object.

(1) Transmission control when the function of the bulk transmission is valid.
(a) Segment process for the ASL-SDU.
The ASL-ELCP shall perform the following process to a mobile station with the valid bulk transmitting function in the ASL mobile station profile.

When the size of the ASL-SDU is smaller than the SUU, the ASL-ELCP adds the communication control information of the invalid bulk transmission mode to the acquired ASL-SDU and generates the ASL-PDU according to the rules in sub-clause 3.2.2.3.1. And the ASL-PDU is set to the transmission queue.

When the size of the ASL-SDU is larger than the SUU, the ASL-SDU is divided into bulk segment by the value of SUU sequentially from a head. And the ASL-ELCP adds the communication control information to each bulk segment, and the ASL-ELCP generates the ASL-PDU according to the rules in sub-clause 3.2.2.3.1, and sets it to the transmission queue.

Figure 3.2-7 shows the example in which the ASL-ELCP adds the communication control information to the ASL-SDU, which is passed from the ASL-NCP.

(b) Assignment of the slots
When the bulk transmission is performed, the bulk segment shall be assigned to the communication slot of the DSRC in consideration of adjustment of a multi-link, etc.

The method of the slot assigning is shown in Annex G; however the procedure is not specified because it is not necessary to take into consideration from a viewpoint of interconnection.

(c) Band control
When the ASL-ELCP performs the bulk transmission, the ASL ELCP shall perform the

control in consideration of the communication band of the assigning to the mobile station.

The method of the band control is shown in Annex G; however the procedure is not specified because it is not necessary to take into consideration from a viewpoint of interconnection.

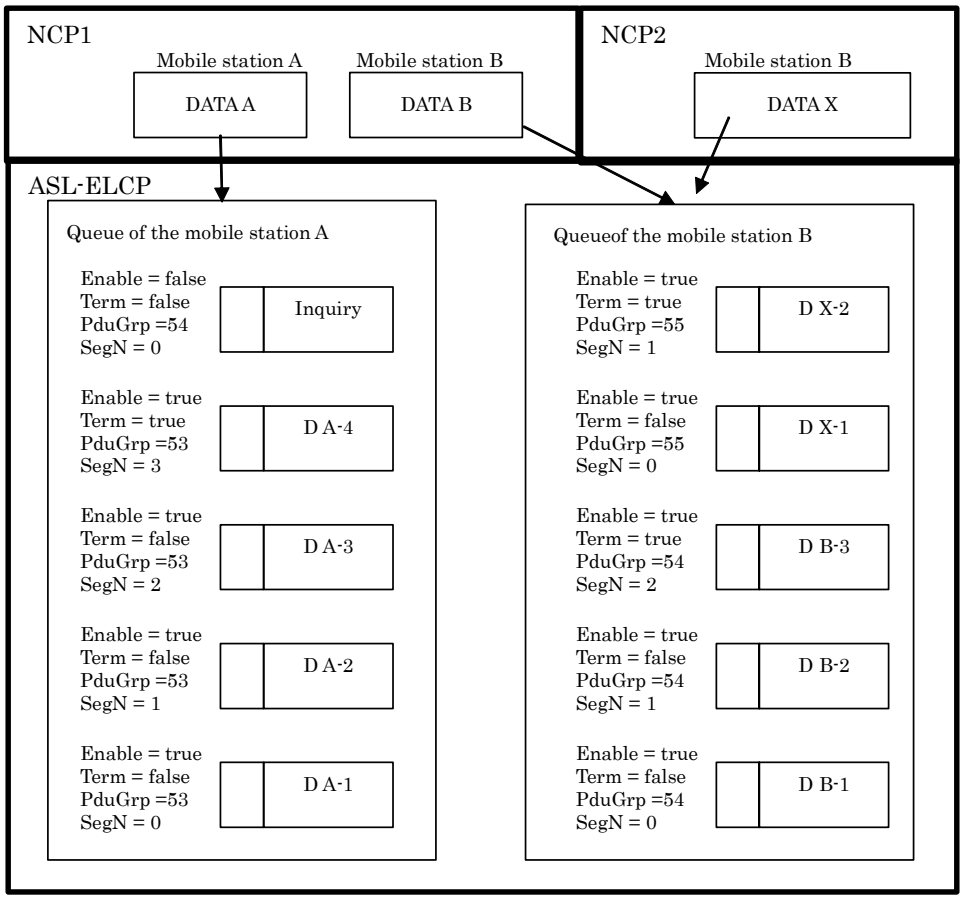


Figure 3.2-7 — Example of the Communication Control Information Addition

(2) Transmission control when the function of the bulk transmission is invalid
The ASL-ELCP shall not apply the processing of the bulk transmission to a mobile station, which the bulk transmitting function invalidates in the ASL mobile station profile.

In this case, the ASL-ELCP shall generate the communication control information, which is shown that the segmentation processing is not performed according to the rules in sub-clause 3.2.2.3.1, and add it to the acquired ASL-SDU. And the ASL-ELCP shall store it to the transmission queue.

3.2.2.4.2.2 Bulk Transmission Process of the Mobile Station

In the bulk transmission processing of the mobile station, the following processes shall be performed with reference to the communication control information in the ASL-PDU stored in the receiving queue.

When the bulk transmission valid identifier in the bulk segments is true and all segments are the same PDU group number, and all ASL-Plus from the segment of which segment number is “0” to the segment of which the bulk transmission termination identifier is true are stored in the receiving queue, the ASL-ELCP connects the Plus in order of a segment number and reproduces the ASL-SDU.

When the bulk transmission valid identifier is false, the ASL-ELCP deletes the communication control information from the ASL-PDU, and reproduces the ASL-SDU.

3.2.2.4.3 Broadcast Mode Control Procedure

3.2.2.4.3.1 Base Station Setup

When the base station provides the service of the broadcast mode only, the base station shall not assign the ACTS.

3.2.2.4.3.2 Broadcast Mode Control Range

Only the ASL-NCP of the mobile station, which corresponds to the ASL-NCP of the base station, shall be able to receive the data transmitted in the broadcast mode control.

3.2.2.4.3.3 Broadcast Mode Control Process of the Base Station

When the link address in the ASL-SDU is the link address of the multicast, the ASL-ELCP shall perform the following processes using the broadcast mode control.

(1) Transmission process when the base station supports the bulk transmission control

The ASL-ELCP segments the ASL-SDU into the ASL-PDU according to the procedure of the bulk transmission process of the base station shown in sub-clause 3.2.2.4.2.1(1), and sets it to the transmission queue of the broadcast. However, the SUM is applied to the unit of segmentation.

On this occasion, the PDU group number of the communication control information, which is added to the segment assign the consecutive numbers by modulo 32. And the option field of the communication control information is valid and the auxiliary parameter for broadcast mode shall be added to the communication control information of the bulk segment.

Further, the link address of the ASL-PDU is set to the multicast link address value of "0".

(2) Transmission process when the base station does not support the bulk transmission control

The ASL-ELCP generates the ASL-PDU according to the procedure of the invalid bulk transmission processing shown in sub-clause 3.2.2.4.2.1(2), and sets it to the transmission queue of the broadcast.

On this occasion, the PDU group number of the communication control information, which is added to the segment assign the consecutive numbers by modulo 32. And the option field of the communication control information is valid and the auxiliary parameter for broadcast mode is added to the communication control information of the bulk segment.

Further, the link address of the ASL-PDU is set to the multicast link address value of "1".

(3) Transmitting repetition process

All ASL-Plus stored in the transmission queue for the broadcast are transmitted in order of the PDU group number; this processing is repeated in times of "k".

When the k time's repetitions are completed, all segments of this PDU group number are discarded.

(4) Hybrid process with the point-to-point communication

In case of coexistence with the point-to-point communication, the slot for the broadcast mode control is assigned to one slot or more in one frame.

In case of coexistence, the control methods of the assignment of the slot and the band control are shown in Annex G, however the procedure is not specified because it is not necessary to take into consideration from a viewpoint of interconnection.

3.2.2.4.3.4 Broadcast Mode Control Process of the Mobile Station

The mobile station performs the broadcast mode control processing to the ASL-PDU stored in the receiving queue of the broadcast.

(1) Management of the validity term on the broadcast mode control

When the mobile station is the non-connected communication state, the mobile station shall manage the validity term of the broadcast service by using to the parameter “serviceTime” which is the auxiliary parameter of the broadcast mode in the communication control information of the ASL-PDU. This validity term shall be managed using the function of the communication connection management in the communication control management (Refer to sub-clause 3.2.3.3.2.3).

The mobile station is the non-connected communication state, and with the first arrival of the ASL-PDU of the multicast link address, the communication control management shall start the CTO which is set to T1max of the parameter “serviceTime” passed by the ASL-PDU (refer to sub-clause 3.2.3.3.1.2.1).

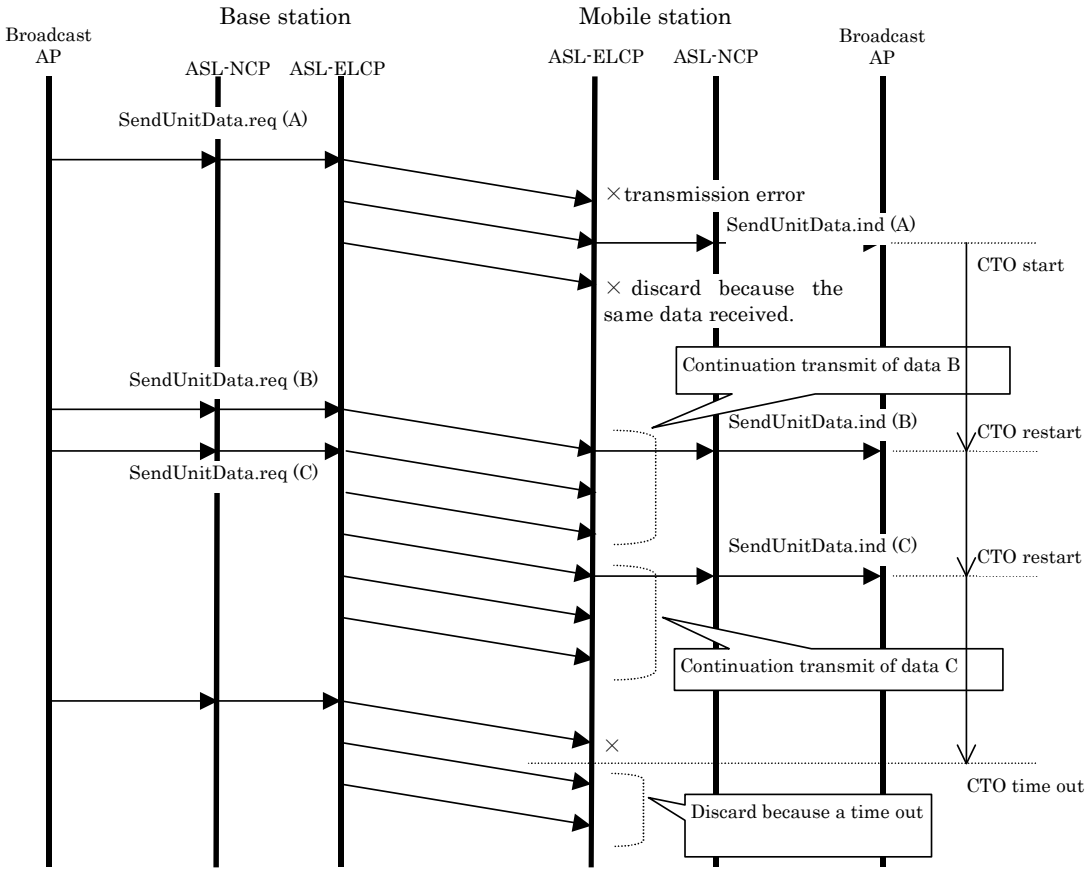


Figure 3.2-8 — Example of the Validity Term Management in the Broadcast Mode Control

Henceforth, whenever the communication control management of the mobile station receives the notice of the arrival of a valid service primitive from a base station, it shall re-start the CTO.

When the CTO expired because of there is no arrival of an effective service primitive from the base station, the communication control management shall be recognized to be the end of the validity term of the broadcast service, and the control management shall discard all PDUs

in the receiving queue for the broadcast service, and await the new notice of connection, or the broadcast service receiving.

The example of the management procedure of the validity term on the broadcast mode control according to the above procedure is shown in Figure 3.2-8.

NOTE When the point-to-point communication is performed in parallel with broadcast communication, the mobile station uses the timer value given by the ASL base station profile.

(2) Reception process when the mobile station supports the bulk transmission control
In processing of broadcast mode control of the mobile station, when the mobile station supports the bulk transmission control, the following processes shall be conducted according to the communication control information in the ASL-PDU stored in the receiving queue for the broadcast communication.

However, the ASL-PDU of the multicast link address value of “0” or “1” is processed, and the ASL-PDU, which does not satisfy this requirement, shall be discarded.

When the bulk transmission valid identifier indicates the false, the ASL-ELCP deletes the communication control information from the ASL-PDU, and reproduces the ASL-SDU.

When the bulk transmission valid identifier in the bulk segments is true and all segments are the same PDU group number, and all ASL-PDUs from the segment of which segment number is “0” to the segment of which the bulk transmission termination identifier is true are stored in the receiving queue, the ASL-ELCP shall unity the PDUs in order of a segment number and reproduce the ASL-SDU.

After uniting reproduces, the processing shall not conduct until the bulk segment of a different ASL-PDU group number is received. After the bulk segment of a different ASL-PDU group number is received, the processing shall be re-started according to the content of communication control information. When the bulk segment with a same PDU group number is received while reserving processing, the PDU shall be discarded.

When the bulk segment with a different PDU group number is received while processing the uniting reproduction, the processing under execution shall be discarded, and the uniting processing shall be begun for a new PDU group number.

(3) Reception processing when mobile station does not support the bulk transmission control
In processing of the broadcast mode control of the mobile station, when the mobile station does not supports the bulk transmission control, the following processes shall be conducted according to the communication control information in the ASL-PDU stored in the receiving queue for the broadcast communication.

However, the ASL-PDU of the multicast link address value of “1” is processed, and the ASL-PDU, which does not satisfy this requirement, shall be discarded.

When the bulk transmission valid identifier indicates false, the ASL-ELCP deletes the

communication control information from the ASL-PDU, and reproduces the ASL-SDU.

When the bulk transmission valid identifier indicates true, the ASL-ELCP should discard the ASL-PDU.

3.2.2.4.4 Client / Server Communication Control

The Extended link control protocol performs the procedure of the Client / Server communication control shown in Figure 3.2-9 in order to be able to enable a transmission from the mobile station. This communication control procedure is shown below.

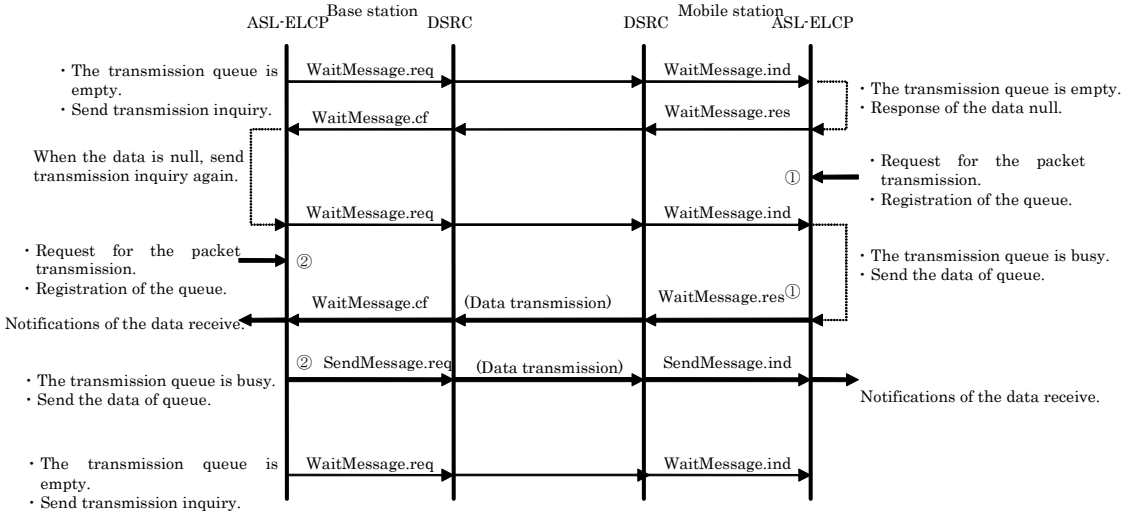


Figure 3.2-9 — Example of Client / Server Communication Control using ACTION Primitive

3.2.2.4.4.1 Base Station Communication Control

(1) Transmission of the ASL-PDU
When there is the ASL-PDU to the mobile station in the transmission queue, the base station transmits the ASL-PDU to the mobile station by using the data transfer service or the data transmission inquiry service.

(a) Transmission of the ASL-PDU by the data transmission service
The ASL-ELCP shall set the decided value to each parameter of the "ACTION.request" primitive according to the specification of the "SendMessage.request" sub-primitive. And the ASL-ELCP shall transmit the ASL-PDU on the data transfer service by using this ACTION primitive of the DSRC layer 7.

In this case, when the link address in the ASL-PDU is the private link address, the value of the eid passed from the DSRC layer 7 by "NotifyApplicationRSU" shall be used for the value of eid in the "ACTION.request" primitive. And when the link address is the multicast link address, the parameter eid in the "ACTION.request" primitive shall be set to

"3" according to regulations of the ARIB STD-T75.

(b) Transmission of the ASL-PDU by the data transfer inquiry service (optional)
The ASL-ELCP sets the decided value to each parameter of the "ACTION.request" primitive according to the specification of the "WaitMessage.request" sub-primitive. And the ASL-ELCP transmits the ASL-PDU on the data transfer inquiry service by using this "ACTION primitive" of the DSRC layer 7.

However, when the link address of the ASL-PDU is a private link address only, using this primitive transmits the ASL-PDU.

In addition, this transfer function of the ASL-PDU is an option. When the function is implemented, it may be selected.

(2) Reception the ASL-PDU (the data transmission inquiry)
(a) Transmission of inquiry
The base station individually inquires the data transmission of the mobile station according to the schedule decided beforehand.

The ASL-ELCP shall set the decided value to each parameter of the "ACTION.request" primitive according to the specification of the "WaitMessage.request" sub-primitive. And the ASL-ELCP shall use the "ACTION.request" primitive of the DSRC layer 7, and perform the transfer inquiry for each mobile station.

The content of the parameter "dataPacket" in the "WaitMessage.request" is usually NULL (data length is "0"). When the option in sub-clause 3.2.2.4.4.1(a) is selected and there are ASL-PDUs to the mobile station in the transmission queue, the ASL-PDUs shall be transmitted by using this primitive.

The link address, which is used in this "WaitMassege.request", shall be a private link address only.

The schedule of a concrete transfer inquiry is not provided for as a practical realization requirement. But the base station shall complete the transmission of the inquiry at least once for those mobile stations until the transmission schedule timer of each mobile station ends. (WTTS: Watchdog Timer for Transmission Schedule, refer to sub-clause 3.2.3.3.1.2.2 and sub-clause 3.2.3.3.2.3).

(b) Reception of response to inquiry
The base station acquires the ASL-PDU, which is transmitted by the mobile station using the "WaitMessage.confirm" sub-primitive.

When the content of the "ACTION.confirm" passed from the DSRC layer 7 satisfies the requirement of the "Waitmessage.confirm" sub-primitive, the base station shall receive this sub-primitive. When the content of the "ACTION.confirm" does not satisfy the requirement, the base station shall discard this primitive.

Next, the ASL-ELCP shall confirm the content of the parameter “dataPacket” primitive, which is received. As a result of the content confirmation, when the content of the parameter “dataPacket” is not NULL (data length is “0”) and the size of the “dataPacket” is not larger than the value in which the MRU of the ASL-ELCP is added to the size of the communication control information, the content of the parameter “dataPacket” shall be stored to the receiving queue corresponding to the link address.

The sub-primitive of which the content confirmation ended shall be discarded one by one.

3.2.2.4.4.2 Mobile Station Communication Control

(1) Transmission of the ASL-PDU (Response to transmission inquiry)
The mobile station transmits the ASL-PDU using the transfer inquiry service.

When the mobile station receives the “WaitMessage.indication”, the mobile station conducts the response process.

The ASL-ELCP shall set the decided value to each parameter of the “ACTION.response” primitive according to the specification of the “WaitMessage.response” sub-primitive. And the ASL-ELCP shall use the “ACTION.response” primitive of the DSRC layer 7, and perform the response process.

In this case, the content of the parameter “dataPacket” is usually NULL (data length is “0”). When there are ASL-PDUs to the base station in the transmission queue, the ASL-PDU shall be transmitted by using this primitive.

(2) Reception of the ASL-PDU
The mobile station receives the ASL-PDU from the base station using the data transfer service or the data transfer inquiry service.

When the content of the “ACTION.indication” passed from the DSRC layer 7 satisfies the requirement of the “SendMessage.indication” sub-primitive or the “WaitMessage.indication” sub-primitive, the mobile station shall receive this sub-primitive. When the content of ACTION.indication does not satisfy the requirement, the mobile station shall discard this primitive.

Next, the ASL-ELCP shall confirm the content of parameter “dataPacket” in the primitive, which is received. As a result of the content confirmation, when the content of the parameter “dataPacket” is not NULL (data length is “0”) and the size of the “dataPacket” is not larger than the value in which the MRU of the ASL-ELCP is added to the size of the communication control information, the content of the parameter “dataPacket” shall be stored to the receiving queue corresponding to the link address. The sub-primitive of which the content confirmation ended shall be discarded one by one.

When the sub-primitive, which is received from the base station, is the

“WaitMessage.indication” sub-primitive, the procedure in sub-clause 3.2.2.4.4.2(1) shall be performed.

3.2.3 Communication Control Management

3.2.3.1 Management Service Interface

3.2.3.1.1 Primitive Relationship Overview

The communication control management of the ASL-ELCP provides the following management services to the ASL-NCP.

(1) Event notify service
The event notify service provides the following primitive.

EventInformation.indication

To notify the events such as errors, etc. occurred within the ASL-ELCP, the "EventInformation.indication" is passed from the ASL-ELCP to the ASL-NCP of the other party station or its own station.

(2) Echo services
The echo service provides the following primitives.

Echo.request
EchoReply.indication

To request a loop back communication between DSRC-ASLs, the ASL-NCP passes an "Echo.request" to the ASL-ELCP. To indicate the response of the loop back communication between DSRC-ASLs, the "EchoReply.indication" is passed from the ASL-ELCP to the ASL-NCP.

(3) MIB access services
The MIB access service provides the following services.

AslmeGet.request
AslmeGet.indication
AslmeSet.request
AslmeSet.indication

To request acquisition of the MIB parameters of ASL-ELCP, an "AslmeGet.request" is passed from the ASL-NCP to the ASL-ELCP. To notify the acquisition of the MIB parameters of ASL-ELCP, an "AslmeGet.indication" is passed from the ASL-ELCP to the ASL-NCP. To request the setting of the MIB parameters of ASL-ELCP, an "AslmeSet.request" is passed from the ASL-NCP to the ASL-ELCP. To notify the setting result of the MIB parameters of ASL-ELCP, an "AslmeSet.indication" is passed from the ASL-ELCP to the ASL-NCP.

3.2.3.1.2 Service Content Specification

This sub-clause specifies the primitives and parameters related with the management services. The parameters as “interface” (interfacing factors) are described conceptually and the information needed for the receiving end entity is defined. But the detailed realization method to pass this information is not restricted.

The parameter “Link address” identifies the service access point of owns station and the other party station at each ASL-NCP.

The parameter “parameter” passes an ASL-SDU itself or a pointer. Or, this process may be realized by other methods.

The parameter “extensionParameter”, the parameter “echoParameter”, and the parameter “mibParameter” may pass actual data or pointers. Or, these processes may be realized by other methods.

The parameter “status” indicates the status indicating values of events.

The parameter “mibIndex” pass a parameter name, which designates an actual MIB parameter or a pointer. Or, this process may be realized by other methods.

The parameter “mibStatus” indicate a successful or unsuccessful result for the request.

The ASL-ELCP shall handle these parameters passed from the ASL-NCP by the following specifications.

Table 3.2-7 — Management Service Parameters

Name of parameter	ASN.1 type	Remarks
linkAddress	DsrcLID	
Status	MsStatusCode	
extentionParameter	OCTET STRING	
echoParameter	MsEchoParameter	

NOTE Parameters relating with MIB access are specified as internal specifications, therefore they are not be specified as the practical realization requirement.

The logical relationship among management service primitives that the ASL-ELCP provides for the ASL-NCP is shown in Figure 3.2-10.

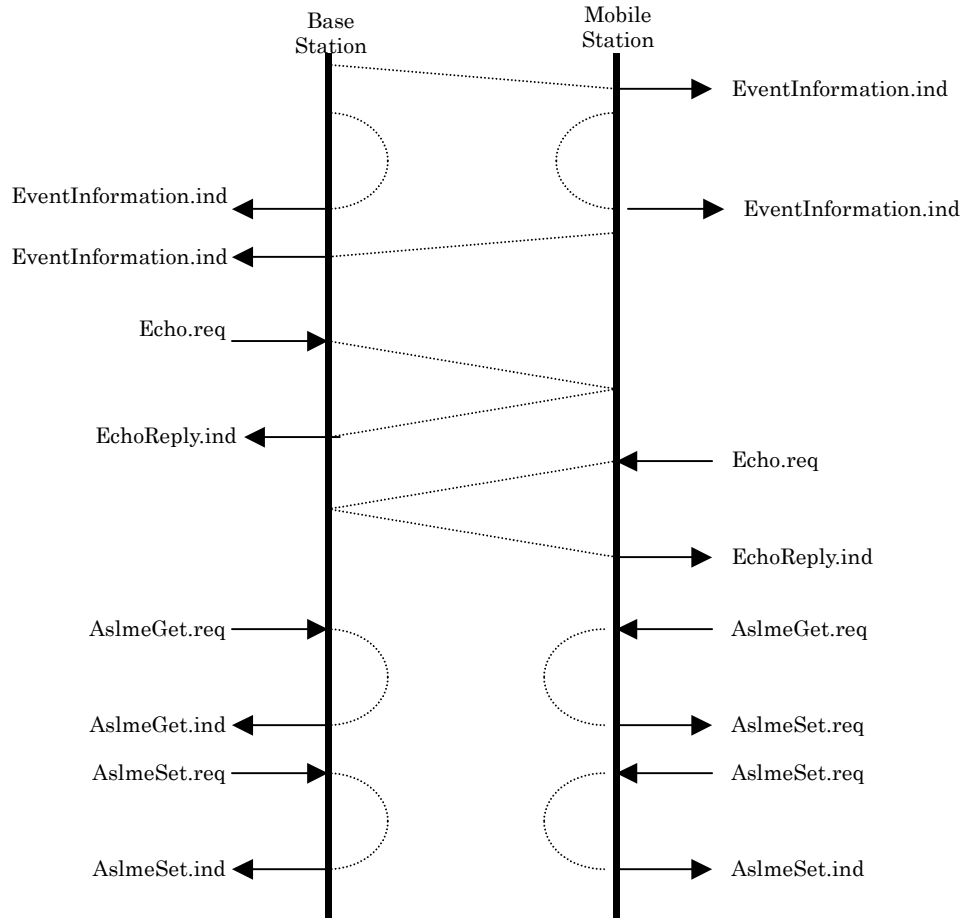


Figure 3.2-10 — Logical Relationship among Management Service Primitives

3.2.3.1.3 Event Notify Services

These services notify events such as errors, etc. occurred within the ASL-ELCP, the ASL-ELCP, and provide the following primitives.

NOTE This primitive process such events occurred within the ASL-ELCP, therefore, the definition of both "PDU coding according to the DsrcControlPDU type in the event notification service" and "the access control information (Member access control according to the DsrcControlPDU type)" is given in the definition of ASL-ELCP.

3.2.3.1.3.1 Event Notify Primitive

(1) Function

This primitive is a service primitive that notifies events such as errors, etc. occurred within the ASL-ELCP.

(2) When generated

The ASL-ELCP generates this primitive when an event such as error, etc. occurred within the ASL-ELCP is notified.

(3) Semantics of the service primitive
This primitive shall provide the following parameter.

EventInformation.indication (linkAddress, status, [extensionParameter])

The parameter “linkAddress” contains the link address to be used in the DSRC.

The parameter “status” contains the code to indicate an event occurred.

The parameter “extensionParameter” contains a piece of information to supplement the contents of the parameter “status” as needed. This parameter is optional.

3.2.3.1.4 Echo Service

This service is a service to perform a loop back communication between DSRC-ASLs, and the ASL-ELCP provides the following service primitives.
NOTE The loop back process within this service shall utilize the function of ASL-ELCP, therefore, the definition of both “PDU coding according to the DsrcControlPDU type in the echo transmission request primitive and “the access control information (Member access control according to DsrcControlPDU type)” shall be given in the definition of ASL-ELCP.

3.2.3.1.4.1 Echo Transmission Request Primitive

(1) Function
This primitive is a service primitive to request a loop back communication between DSRC-ASLs.

(2) When generated
This primitive is always generated by the ASL-NCP.

(3) Semantics of the service primitive
This primitive shall provide the following parameters.

Echo.request (linkAddress, [echoParameter])

The parameter “linkAddress” is contained in the link address to be used in the DSRC.

The parameter “echoParameter” contains an optional data stream to be transferred in the loop back communication. This parameter is optional.

3.2.3.1.4.2 Echo Response Notify Primitive

(1) Function
This primitive is a service primitive to notify a receiving of the response in a loop back

communication between DSRC-ASLs.

(2) When generated
The ASL-ELCP generates this primitive when an echo response arrival is indicated.

(3) Semantics of the service primitive
This primitive shall provide the following parameters.

EchoReply.indication (linkAddress, [echoParameter])

The parameter “linkAddress” contains a link address to be used in the DSRC.

The parameter “echoParameter” contains a data stream that is contained in the parameter “echoParameter” of the “Echo.request” primitive. This parameter is optional.

3.2.3.1.5 MIB Access Service

This service is provided by the MIB management function for communication control management, and performs the setting and the reference of the MIB parameter of ASL-ELCP. The ASL-ELCP provides the following service primitives.

3.2.3.1.5.1 MIB Parameter Acquisition Request Primitive

(1) Function
This primitive shall be a service primitive to request to obtain MIB parameters of the ASL-ELCP.

(2) When generated
The ASL-ELCP generates this primitive always.

(3) Semantics of the service primitive
This primitive shall provide the following parameter.

AslmeGet.request (mibIndex)

The parameter “mibIndex” contains a parameter name that specifies the MIB parameter.

3.2.3.1.5.2 MIB Parameter Acquisition Notify Primitive

(1) Function
This primitive is a service primitive to notify the acquisition of MIB parameters in the ASL-ELCP.

(2) When generated
The ASL-ELCP generates this primitive when the acquisition of MIB parameter is indicated.

(3) Semantics of the service primitive
This primitive shall provide the following parameters.

AslmeGet.indication (mibIndex, mibStatus, [mibParameter])

The parameter “mibIndex” contains a parameter name to specify the acquired MIB parameter.

The parameter “mibStatus” contains the result of a request execution.

The parameter “mibParameters” contains the contents of an acquired MIB parameter. If a designated MIB parameter does not exist or the value isn’t set, this parameter “mibParameter” shall be “omitted”.

3.2.3.1.5.3 MIB Parameter Containing Request Primitive

(1) Function
This primitive is a service primitive to request to set the contents of the MIB parameters in the ASL-ELCP.

(2) When generated
The ASL-NCP always generates this primitive.

(3) Semantics of the service primitive
This primitive shall provide the following parameters.

AslmeSet.request (mibIndex, [mibParameter])

The parameter “mibIndex” contains a parameter name that specifies the MIB parameter.

The parameter “mibParameter” contains the contents of the MIB parameter to be set. At that time, if this parameter is omitted, it means to request to delete the parameter contents.

3.2.3.1.5.4 MIB Parameter Containing Notify Primitive

(1) Function
This primitive is a service primitive to notify the setting result of the MIB parameter of ASL-ELCP.

(2) When generated
The ASL-ELCP generates this primitive when the setting result of the MIB of ASL-ELCP is indicated.

(3) Semantics of the service primitive

This primitive shall provide the following parameters.

AslmrSet.indication (mibIndex, mibStatus)

The parameter “mibIndex” contains a parameter name that specifies the obtained MIB parameter.

The parameter “mibStatus” contains the result that the request was executed.

3.2.3.2 Protocol Data Unit（PDU）

3.2.3.2.1 PDU Format

The communication control management to manage the ASL-ELCP assigns the peer layer access point identifiers as that of the ASL-ELCP in the communication control management in order to formulate the peer layer protocol among management entities, as a result, the data transfer and reception among management entities are possible by utilizing the communication service interface of the ASL-ELCP. For this purpose, the PDU format of communication control management is defined with the network control protocol data unit (NCP-PDU) that is the same as that of the PDU of the ASL-NCP.

The format of NCP-PDU is shown in Figure 3.2-11. The NCP-PDU shall consist of the control field (access control information) that contains the control information to instruct the procedure of ASL-NCP and the information field (network control service unit (NCP-SDU: NCP Service Data Unit)) that contains the PDU of the upper layer protocol.

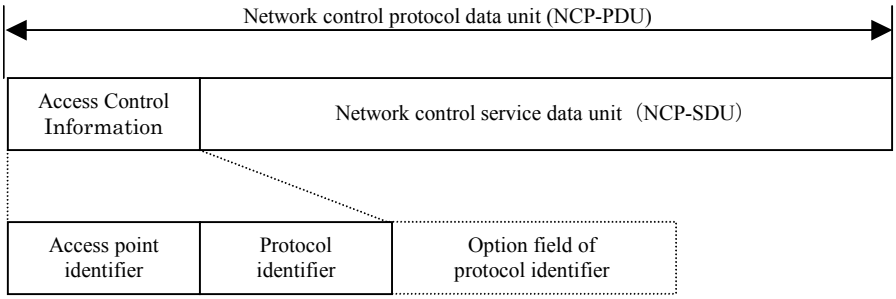


Figure 3.2-11 — Communication Control Management PDU Format

3.2.3.2.2 PDU Element

3.2.3.2.2.1 Connection Identification

The access point identifier to identify the communication control management is given or received in the control field of an NCP-PDU. In addition, the link address to identify connections, etc. is given and received as the service primitive’s parameter that is provided

by the ASL-ELCP.

3.2.3.2.2.2 Control Filed Format

The access control information that is consisted of both access point identifier sub-field and protocol identifier sub-field shall be contained in the control field to indicate the procedure of communication control management.

The field format of the above-mentioned access control information shall be shown in Table 3.2-8. Further, this access control information shall be defined by the parameter “aslAccessProtocol” of the Network Control PDU type that defines the NCP-PDU configuration of the ASL-NCP. (Refer to Annex B.)

Table 3.2-8 — Access Control Information Field Format

	7(MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier				Protocol identifier			
2	Optional field identifier							

(1) Access point identifier sub-field
The value to identify the access point of communication control management shall be contained in the sub-field of access point identifier. The access point identifier of communication control management is specified in Table 3.3-2.

(2) Protocol identifier sub-field
The control information to instruct the procedure of communication control management shall be contained in the sub-field of protocol identifier. In addition, the sub-field of protocol identifier has an optional field for protocol identifier to multiplex additional information that supplements the control contents, and this optional field shall be used for communication control management as needed.

In the communication control management, the identification information shown in Table 3.2-9 is defined and also the messages that are commonly owned by the base station and the mobile station are defined.

Table 3.2-9 — Communication Control Management Protocol identifier

Protocol identifier	Contents	Type of optional field (Refer to Annex B.)
0	Disabled to use	
1	Echo process (Transmission)	MsEchoParameter type
2	Echo process (Response)	MsEchoParameter type
3	Event process	MsEventParameter type
4	Access management (Transmission of random numbers)	MsAuthCodeChallenge type
5	Access management (Response to digital signature)	MsAuthCodeSignature type
6-15	Reserved	

3.2.3.2.2.3 Information Field Format

The information field shall be NULL (data of zero length).

3.2.3.3 Communication Connection Management

3.2.3.3.1 Communication Connection Management Procedure Element

3.2.3.3.1.1 DSRC-ASL Profile Format

The DSRC-ASL profile indicates the characteristics of the DSRC-ASL to the peer station and this profile are composed of the function list that is installed in the ASL-ELCP, the ASL-NCP list, etc.

The configuration definition of the base station DSRC-ASL profile and the mobile station DSRC-ASL profile are specified in Annex B in detail.

3.2.3.3.1.1.1 Base Station DSRC-ASL Profile (RoadSideProfile)

The base station DSRC-ASL profile specified by the “RoadSideProfile” type shall include the following parameters:

(1) Version information (versionIndex)
This information indicates the version information of the DSRC-ASL base station. This information shall be “0” (zero) in this Standard (Version 1.0). Further, the Annex E specifies how to use the version information.

(2) Value of connection control timer for the mobile stations (serviceTime)
This timer value means “T1 max” that a mobile station sets in the CTO that controls the connection status between the mobile station and the base station concerned. The time value shall be set to “0” through “4095” with a millisecond unit. Further, when the time value is set

to “0” (zero), the time length shall be infinite.

(3) ASL-NCP identification information (accessControl)
This information contains an identifier to specify the ASL-NCP type installed in the base station. This identification information shall contain the following identifiers (at least one or more) according to the installed contents. The specification of the identification information is specified in Annex B.

When the communication connection management specified in 3.2.3.3 is installed, the “aslControlManagement” shall be contained. When the LPCP specified in 3.3.3 is installed, the “aslPortControl” or the “localPortControl2” shall be contained according to its usage. When the LANCP specified in 3.3.4 is installed, the “lanControl” shall be contained. When the PPPCP specified in 3.3.5 is installed, the “pppControl” shall be contained.

Further, if the parameter data length does not match with the octet length, the boundary shall be adjusted by adding an “octetAlignment” to the last frame.

(4) Identification information of ASL-ELCP (linkControl) function
This information contains an identifier to identify the function that the ASL-ELCP of the base station supports. The following identifiers shall be contained in this information according to the installed contents. The specification of the information is specified in Annex B.

The identifier “secureFunction” shall be contained when the access control function specified in 3.2.3.4 is installed.

The identifier “bulkTransmit” shall be contained when the bulk transmitting function specified in 3.2.2.4.2 is installed.

The identifier “broadCast” shall be contained when the multicast mode function specified in 3.2.2.4.3 is installed.

3.2.3.3.1.1.2 Mobile Station DSRC-ASL Profile (VehicleProfile)

The following parameters shall be included in the mobile station DSRC-ASL profile specified by the VehicleProfile type.

(1) Version information (versionIndex)
This information indicates the version number of the DSRC-ASL mobile station. This version information shall be “0” (zero) in this Standard, (Version 1.0). Further, Annex E specifies how to use the version information.

(2) Mobile station identification information (equipmentID)
This information is an identifier to identify mobile stations. A unique number shall be assigned to each mobile station in principle. The numbering system, its usage, etc. for this mobile station identification information is defined in Annex F.

(3) ASL-NCP identification information (accessControl)
This information contains an identifier to specify the ASL-NCP type installed in a base station. This identification information shall contain the following identifiers according to the installed contents. The specification of the identification information is specified in Annex B.

When the communication connection management specified in 3.2.3.3 is installed, the "aslControlManagement" shall be contained. When the LPCP specified in 3.3.3 is installed, the "localPortControl" or the "localPortControl2" shall be contained according to its usage. When the LANCP specified in 3.3.4 is installed, the "lanControl" shall be contained. When the PPPCP specified in 3.3.5 is installed, the "pppControl" shall be contained.

Further, if the parameter data length does not match with the octet length, the boundary shall be adjusted by adding an "octetAlignment" to the last frame.

(4) Identification information of ASL-ELCP (linkControl) function
This information contains an identifier to identify the function that the ASL-ELCP of mobile station supports. The following identifiers shall be contained in this information according to the installed contents. The specification of the information is specified in the Annex B.

The identifier "secureFunction" shall be contained when the access control function specified in 3.2.3.4 is installed.

The identifier "bulkTransmit" shall be contained when the bulk transmitting function specified in 3.2.2.4.2 is installed.

The identifier "broadCast" shall be contained when the multicast mode function specified in 3.2.2.4.3 is installed.

3.2.3.3.1.2 Management Control Variables

3.2.3.3.1.2.1 Communication Connection Management Timer (CTR, CTO, T1 max, T2 max)

The communication connection management timer (CTR: Connection Timer for RSU) is the timer to monitor the communication connection status between base station and mobile station. The CTR shall be generated at every communication connection established between a base station and a mobile station and deleted when the communication with the mobile station is terminated.

The CTO is the timer to monitor the communication connection status between a base station and a mobile station. The CTO shall be generated at every communication connection established with the base station and deleted when the CTO expired.

The T1 max is the timer value to be set in the CTO that is passed from a base station in the

form of the parameter “serviceTime” in the ASL base station profile or the parameter “serviceTime” defined in the broadcast transmission mode supplementary parameter in communication control information.

The T2 max shall be the timer value to be set in the CTR.

Further, the values of the T1 max and the T2 max is specified in accordance with the base station installation condition, etc.

3.2.3.3.1.2.2 Watchdog Timer for Transmission Schedule (WTTS)

The WTTS is the timer of a base station to monitor whether the periodical transmission of “Request primitive for transmission inquiry” is ensured for mobile stations.

The WTTS shall be generated at every communication connection established with a mobile station and deleted when the communication with the mobile station is terminated.

The T1 max to be set in the CTO shall be set in the WTTS.

3.2.3.3.1.3 Management Service Interface between the DSRC Layer 7

This subclause specifies the management service interface to be provided by the DSRC layer 7 that is defined in the standard of ARIB STD-T75.

3.2.3.3.1.3.1 Application Management Service

The ASL-ELCP uses the following primitives provided by the DSRC layer 7 to control the use of DSRC environment.

RegisterApplicationRSU (aid, mandatory, priority, eid, [profiles], [parameter])
RegisterApplicationOBU (aid, priority, eid, [profiles], [parameter])
DeregisterApplication (aid, [eid])

The ASL-ELCP of a base station passes the “RegisterApplicationRSU” to the DSRC layer 7 of the base station for the purpose of that the DSRC-ASL of the base station requests the use of DSRC environment. The ASL-ELCP of a mobile station passes the “RegisterApplicationOBU” to the DSRC layer 7 of a mobile station for the purpose of that the DSRC-ASL of the mobile station requests the use of DSRC environment.

The ASL-ELCP of a base station and a mobile station pass the “DeregisterApplication” to the DSRC layer 7 in order to request the termination of the use of DSRC environment.

The specification concerning the management service of the DSRC layer 7 is shown as follows:

Table 3.2-10 — RegisterApplicationRSU Parameter Setting

Parameter name	Value	Remarks
Aid	18	
Mandatory	A value to indicate mandatory or non-mandatory	As per base station specification
Priority	An optional value	As per base station specification
Eid	Omitted	
Profile	The values of more than one out of 10, 11, and 12	As per base station specification
Parameter	Contains an ASL mobile station's profile	

Table 3.2-11 — RegisterApplicationOBU Parameter Setting

Parameter name	Value	Remarks
Aid	18	
Priority	An optional value	As per base station specification
Eid	the value out of 4 through 127	Duplication shall not be permitted.
Profile	The values of more than one out of 10, 11, and 12	As per base station specification
Parameter	Contains an ASL mobile station's profile	

Table 3.2-12 — DeregisterApplication Parameter Setting

Parameter name	Value	Remarks
Aid	18	
Eid	-	Omitted

The base station DSRC-ASL Profile

parameter Container ::= octetstring: aslParameter

--aslParameter shall be the value with a name that shows the result of encoding by RoadSideProfile type. Refer to the Annex B for the details of RoadSideProfile type.

The mobile station DSRC-ASL Profile

parameter Container ::= octetstring: aslParameter

--aslParameter shall be the value with a name that shows the result of encoding by VehicleProfile type. Refer to the Annex B for the details of VehicleProfile type.

Figure 3.2-12 — DSRC-ASL Profiles Containing

3.2.3.3.1.3.2 Communication Management Service

The ASL-ELCP uses the following primitives that the DSRC layer 7 provides for the communication connection management.

NotifyApplicationRSU (priority, [eid], lid, [parameter], obeConfiguration)
NotifyApplicationOBU (beacon, priority, [eid], lid, [parameter])
EndApplication (eid, lid, [norm_end])
NotifyApplicationOBU_Release ( )

The "NotifyApplicationRSU" is passed from the DSRC layer 7 of a base station to the ASL-ELCP of the base station to notify that the communication connection with the mobile station completed. The "NotifyApplicationOBU" is passed from the DSRC layer 7 of the mobile station to the ASL-ELCP of the mobile station to notify that the communication connection with a base station completed. The ASL-ELCP passes an "EndApplication" to the DSRC layer 7 for terminating the communications between DSRC-ASLs and for releasing the peer connection.

The "NotifyApplicationOBU_Release" is passed from the DSRC layer 7 of the mobile station to the ASL-ELCP of the mobile station to notify that a base station disconnects the connection.

The following table shows the specification concerning the communication management service of the DSRC layer 7.

Table 3.2-13 — NotifyApplicationRSU Primitive Parameter Setting

Parameter name	Value	Remarks
Priority	A value generated by a base station's layer 7	
Eid	A value generated by an mobile station	
Lid	A link address generated by an mobile station	
Parameter	A mobile station DSRC-ASL profile	
obeConfiguration	A value of Parameter "obeConfiguration" in VST	

Table 3.2-14 — NotifyApplicationOBU Primitive Parameter Setting

Parameter name	Value	Remarks
Beacon	A value of Parameter "rsu" in BST	An identifier of base station
Priority	A value generated by a base station's layer 7	
Eid	-	Omitted
Lid	Its own address of an mobile station generated by itself	
Parameter	A base station DSRC-ASL profile	

Table 3.2-15 — EndApplication Primitive Parameter Setting

Parameter name	Value	Remarks
EID	A value passed by “NotifyApplication RSU”, or A value passed by “RegisterApplicationOBU”	
LID	A value passed by “NotifyApplication RSU”, or A value passed by “RegisterApplicationOBU”	
Norm_end	Normal ending (1)	Option

3.2.3.3.2 Communication Connection Management Procedure

3.2.3.3.2.1 Service Requirement of DSRC Environment

The ASL-ELCP shall be enable to use a DSRC environment by using an application management service provided by the DSRC layer 7 in principles in case one or more usable ASL-NCPs exist after the ASL-ELCP is activated.

But, regarding to a mobile station even when a usable ASL-NCP does not exist, the ASL-ELCP shall be enable the mobile station to use a DSRC environment.

3.2.3.3.2.2 Communication Connection Procedure

A base station and a mobile station shall perform the following communication connection procedure when a communicable version is selected based upon the version information of a DSRC-ASL profile.

The communication connection procedure of a base station and mobile stations are shown in the following. In addition, Figure 3.2-13 shows an example of the procedure of the base station and mobile stations specified in this subclause.

(1) Communication connection procedure of the base station
When the ASL-ELCP receives an initial connection notice by a primitive “NotifyApplicationRSU”, the ASL-ELCP refers to the function identification information of the ASL-ELCP from the ASL mobile station profile contained in the parameter “parameter”. And, the base station compares its own function to the mobile station function; the base station judges that only functions owned by both units commonly are effective; and the base station enables to conduct the procedure defined in each function.

Further, the base station refers to the ASL-NCP identification information in the mobile station DSRC-ASL profile, And, the base station compares its own ASL-NCP to the mobile station one, it enables an corresponding ASL-NCP to be usable for communications with the mobile station concerned, and it shall issue a status “Communication connection notice” by using an event notice primitive in the management service to the ASL-NCP concerned. But, if the access management function (Refer to 3.2.3.4) is effective, this procedure is conducted after the equipment authentication was successful.

If a usable ASL-NCP does not exist, the base station terminates the communication with the mobile station after issuing an “EndApplication” primitive. But, if a usable LPCP exists in the base station, the base station issues both a status “Communication connection notice” and a status “Communication disconnection notice” to the LPCP one by one by using an event notice primitive in the service management and the base station terminates the communication with the mobile station.

(2) Communication connection procedure of the mobile station
When the ASL-ELCP receives an initial connection notice by a primitive “NotifyApplicationOBU”, the ASL-ELCP shall refer to the function identification information of ASL-ELCP from the ASL base station profile contained in the parameter ‘parameter’. And, the mobile station compares its own function to the base station function; the mobile station shall judge that only functions owned by both units commonly are effective; and the mobile station enables to conduct the procedure defined in each function.

Further, the mobile station refers to the ASL-NCP identification information in the base station DSRC-ASL profile, it enables an corresponding ASL-NCP to be usable for communications with the base station concerned, and it issues a status “Communication connection notice” by using an event notice primitive in the management service to the ASL-NCP concerned. But, if the access management function (Refer to 3.2.3.4) is effective, this procedure is conducted only when the equipment authentication was successful.

If a usable ASL-NCP does not exist, the mobile station terminates the communication with the base station after issuing an “EndApplication” primitive.

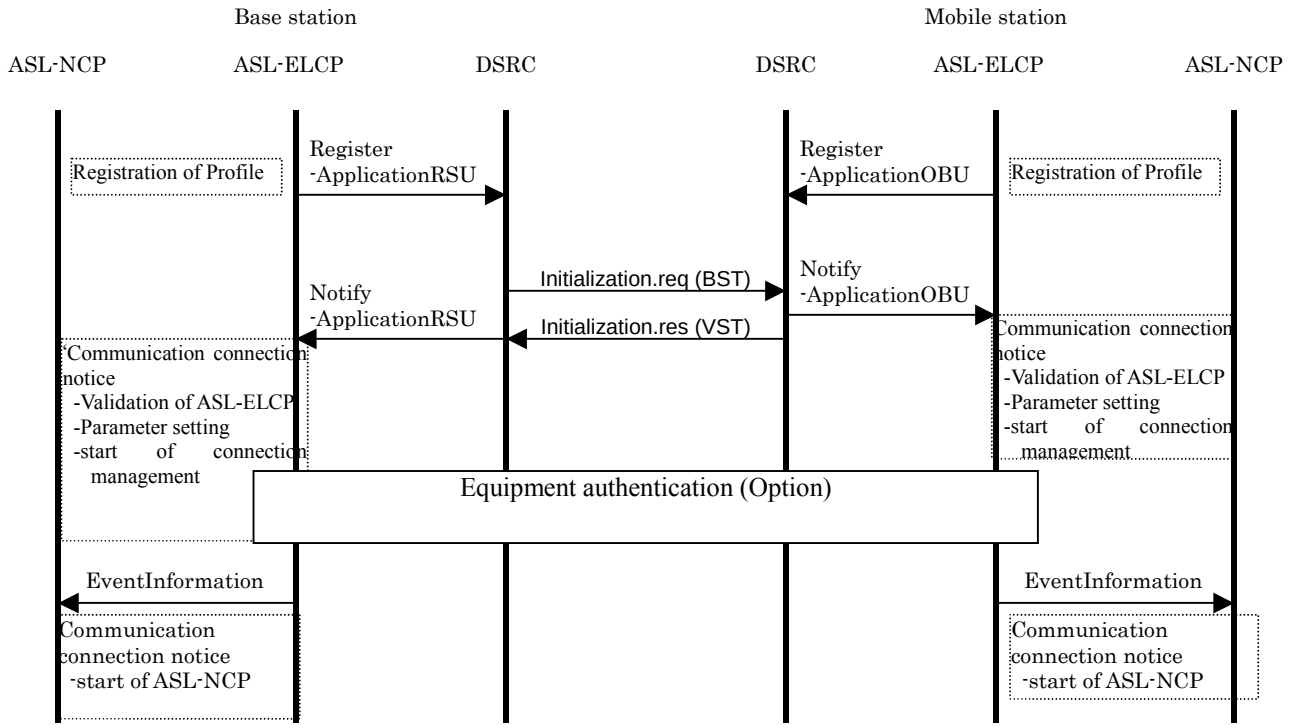


Figure 3.2-13 — Example of Communication Connection Procedure

3.2.3.3.2.3 Communication Connection Management Procedure

The procedure to maintain and manage the communication management of a base station and mobile stations is described in the following. In addition, Figure 3.2-14 shows the connection management procedure of base station and mobile stations specified in this subclause.

(1) Communication connection management procedure of the base station
In the communication control management of base station, the CTR and the WTTS shall be generated for each mobile station at the beginning of communication connections. At that time, T2 max shall be set in the CTR and T1 max set in the WTTS.

In the communication control management of the base station, the CTR and the WTTS shall be started triggered by the transmitting of a “Request primitive of data transmission inquiry” to manage the communication connection status with a mobile station.

The CTR shall be stopped when receiving a “Response primitive of data transfer inquiry” that corresponds to the above request primitive.

The WTTS shall be restarted when transmitting a “Request primitive of the next data transfer inquiry”.

When a timeout of the CTR occurs owing to a receiving-disable of the “Response primitive

of data transfer inquiry", or when a timeout of the WTTS occurs owing to a transmission disable of "Request primitive of data transmission inquiry", a status "Communication disconnection notice" shall be issued to the ASL-NCP being enabled by using an Event notice primitive of management service in order to terminate the use of ASL-NCP for the mobile station concerned. Further, the processes, the CTR, and the WTTS performed with the onboard station shall be terminated after issuing an "EndApplication" primitive.

(2) Communication connection management procedure of the mobile station
In the communication connection management of mobile station, the CTO shall be generated at the beginning of communication connections. By referring to the communication connection timer value of a mobile station from the base station's profile contained in the parameter of a "NotifyApplicationOBU" primitive, the timer value shall be set in the CTO and shall be started.

After this process onward, the CTO shall be restarted every time when a valid service primitive arrival from the base station is notified.

When a timeout of the CTR occurs owing to no receiving of a valid service primitive, a status "Communication disconnection notice" shall be issued to the ASL-NCP being enabled by using an "event notice" primitive of management service in order to terminate the use of the ASL-NCP for the mobile station concerned.

Further, the processes and the CTO performed with the base station shall be terminated after issuing an "EndApplication" primitive, and a new connection notice or waiting for broadcast receiving shall be enabled to conduct.

In addition, when a "NotifyApplicationOBU_Release" is received at a certain period during a startup operation of the CTO, the CTO shall be terminated, and a status "Communication disconnection notice" shall be issued to the ASL-NCP being enabled by using an Event notice primitive of management service in order to terminate the use of the ASL-NCP for the mobile station concerned. Further, the processes with the base station shall be terminated after issuing an "EndApplication" primitive and a new connection notice or waiting for broadcast receiving shall be enabled to conduct.

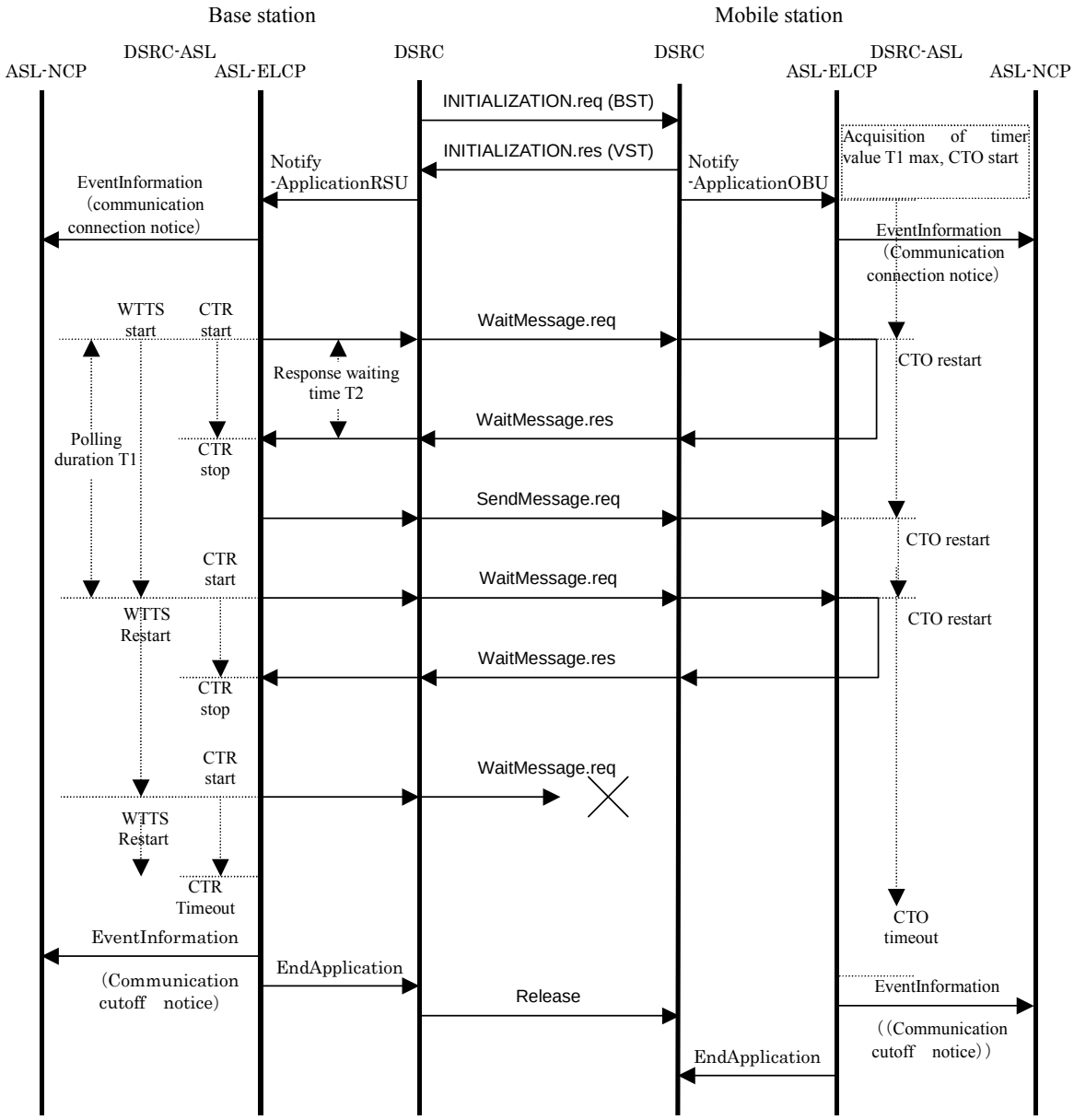


Figure 3.2-14 — Example of Communication Connection Management Procedure

3.2.3.4 Access Control

3.2.3.4.1 Access Management Procedure Elements

3.2.3.4.1.1 Security Profile

When a security function is registered in the function identification information of the ASL-ELCP in the DSRC-ASL profile, this profile is valid information.

In the access management, the setting of option information, etc. is performed by referring to

the security profile when the equipment authentication function is effective.

The configuration of the security profile is defined as the ProfileSecurity type. (Refer to Annex B)

(1) Equipment authentication valid identifier (authenticate)
This identifier is an identifier to indicate a valid or invalid status of the equipment authentication function.

When this identifier indicates a true value, the equipment authentication shall be performed. When the equipment authentication is not performed, this identifier shall be a false value.

(2) Equipment identifier (userID)
This identifier is an identifier to use for the equipment authentication. This identifier shall be used only for the mobile station. Null (data whose length is zero) shall be contained for the base station.

(3) Equipment authentication algorism identifier (authenticationMethod)
This identifier is a digital signature algorism identifier for equipment authentication. (Optional)

When the digital signature algorism is specified, the algorism identifier shall be contained.

Further, when the equipment authentication algorism identifier doesn't exist, a default digital signature algorism shall be used.

NOTE It is assumed that the system operator assigns the equipment authentication algorism identifier.

3.2.3.4.1.2 Default Digital Signature Algorism

The default digital signature algorism of this standard version is as shown in the following:

When a default digital signature algorism is selected, the password written in common word shall be adopted as the calculation result by not performing a digital signature calculation with random numbers and passwords.

3.2.3.4.1.3 Number of Authentication Trials (NA)

The number of authentication trials (NA) is the maximum number that a mobile station is allowed to retry the authentication request. The base station shall admit the retry only when the number of trials from the identical mobile station doesn't exceed the NA. If exceeded the NA, the communication shall be disconnected. Further, the NA value is not specified as a practical installation requirement.

3.2.3.4.1.4 Access Control Information Format

The control information to specify the procedure for access management is contained in the access management and the message possessed commonly by a base station and a mobile station is defined in the access management.

3.2.3.4.1.4.1 Transmission Message of Random Numbers

This message is the message to transmit a random number that configures the digital signature base data. The transmission message format of random numbers shall be shown in Table 3.2-16.

Table 3.2-16 — Random Numbers Transmission Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier linkControlManagement (0)				Protocol identifier challenge (4)			
2	Length of random number							
3	Contents of random number							
	:							

(1) Access point identifier
This identifier contains an identifier “linkControlManagement” (0) which indicates the communication control management.

(2) Protocol identifier
This identifier contains an identifier “challenge” (4) which indicates the transmission message of random numbers.

(3) Option field of protocol identifier
This identifier contains the following contents as a result of encoding with an MsAuthCodeChallenge type.

(a) Random number length identifier
This identifier shall indicate the data length of the succeeding random number. The unit shall be octet.

(b) Random number contents
The random number shall be of a parameter length data of “0” through “255” octets.

3.2.3.4.1.4.2 Response Message for the Digital Signature

This message shall be transmitting the digital signature result for the received base random

number. Table 3.2-17 shows the response message format for digital signature.

Table 3.2-17 — Response Message Format for Digital Signature

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier linkControlManagement (0)				Protocol identifier signature (5)			
2	Length of digital signature							
3	Contents of digital signature							
	:							

(1) Access point identifier
This identifier contains an Identifier “linkControlManagement” (0) to indicate the communication control management.

(2) Protocol identifier
This identifier contains an Identifier “signature” (5) to indicate the response message of digital signature.

(3) Option field of protocol identifier
This identifier contains the following contents as a result of encoding with an MsAuthCodeSignature type.

(a) Digital signature length identifier
This identifier shall indicate the data length of the succeeding digital signature. The unit shall be octet.
(b) Digital signature contents
The digital signature shall be of a parameter length data of “0” through “255” octets.

3.2.3.4.2 Access Management Procedure

Figure 3.2-15 shows the outline of equipment authentication procedure.

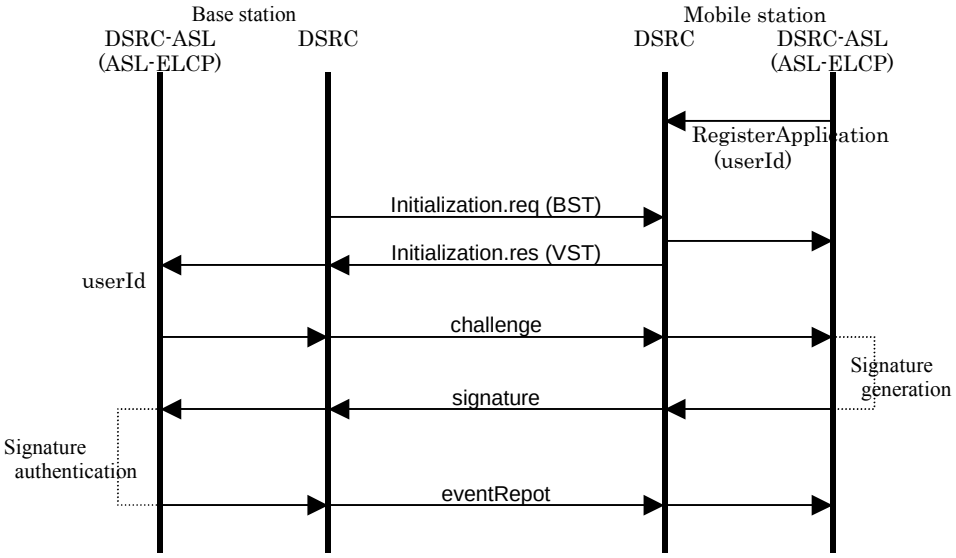


Figure 3.2-15 — Outline of Equipment Authentication Procedure

3.2.3.4.2.1 Initial Setup in Equipment Authentication

(1) Initial setting of the base station
When a base station receives instructions by a "NotifyApplicationRSU" primitive from the DSRC layer 7, the base station shall move its status to the "Equipment authentication process". At that time, the contents of security profile in the ASL mobile station profile shall be checked by the following procedure.

(a) When the value of the equipment authentication valid identifier is a true one, the content of the digital signature algorism identifier is checked to select an algorism. At that time, if the digital signature algorism identifier does not exist, a default algorism is selected.

(b) Next, a password corresponding to the identifier concerned is obtained by referring to the equipment identifier, and the status moves to the authentication process shown in 3.2.3.4.2.2 (1). Further, the acquisition method of password is not specified.

(c) When the security profile of the mobile station does not exist and if the equipment authentication valid identifier value is a false one, the equipment authentication process is terminated.

NOTE1 If the contents of digital signature algorism identifier does not correspond with a mobile station one, or when the value of the "equipment authentication valid identifier" is a true one, the process is be a matter of practical application whether the process selects a default algorism or goes to terminate the communication. Such practical applications are not defined here.
NOTE2 The matter whether or not the access to a base station from a mobile station not equipped with the equipment authentication function may accept is a matter of practical application. Such practical applications are not defined here.

(2) Initial setup of the mobile station
When a mobile station receives instructions by a “NotifyApplicationOBU” primitive from the DSRC layer 7, the mobile station shall move its status to the “Equipment authentication process”. At that time, the contents of security profile in the base station DSRC-ASL profile shall be checked by the following procedure.

(a) When the value of the equipment authentication valid identifier (authenticate) is a true one, the contents of the digital signature algorism identifier is checked to select an algorism. At that time, if the digital signature algorism identifier does not correspond with the mobile station one or doesn’t exist, a default algorism is selected.
(b) Upon completion of the algorism selection, the process moves to the authentication process shown in 3.2.3.4.2.2 (2).

3.2.3.4.2.2 Equipment Authentication Process

(1) Equipment authentication process in the base station
Upon transiting to a status “Equipment authentication process”, a base station shall generate a random number first. The generated random number shall be transmitted to a mobile station by a random number transmission message and the base station shall wait its response. In addition, the generated random number shall be signed with a password obtained by the initial setting procedure and a digital signature data shall be composed.

When the base station receives a response by the digital signature response message, the base station shall authenticate the mobile station by comparing “the digital signature date contained in the message” with “the digital signature data composed by the base station”.
When the comparison result coincides with each other, the authentication shall be recognized as successful, and the base station shall notify the mobile station that the access to the base station was allowed with an event notice message (Refer to 3.2.3.5.) A status “Access was allowed” shall be set in the status identifier “status”. When the comparison result does not coincide, the authentication shall be recognized as unsuccessful, and the base station shall notify the mobile station that the access to the base station was denied with an event message. At that time, a status “Access was rejected” shall be set in the status identifier “status”.

When the received message contents are different from an expected value, the message shall be discarded, and the result shall be counted in the number of failed authentications and a new message arrival shall be waited. Further, if a time-out of the CTR or the WTTS occurs, the process contents shall be cancelled and the communication shall be cut off to terminate it. If the number of failed authentications for the mobile station is less than the number of the NA, the communication retry shall be permitted and a random number shall be regenerated and transmitted with transmitting message. If the number of failed authentication is over the NA, the communication shall be disconnected immediately.
NOTE When a mobile station which communication was cutoff once owing to a failed authentication is now allowed to retry communication-connections, appropriate countermeasures is provided against the risk that unlimited accesses might occur with high frequency.

(2) Equipment authentication process in the mobile station

When a mobile station receives a random number transmission message, the mobile station shall perform a digital signature process with the password set to the mobile station based upon the random number contained in the random number transmission message as the digital signature base data. The digital signature data obtained by the digital signature process shall be contained in a digital signature response message and transmitted to the base station, and the mobile station shall wait for the authentication result.

When a notice of "Authentication successful" is received by an event notice message (Refer to 3.2.3.5.), the access point with the ASL-NCP shall be enabled and the authentication process shall be terminated. When a notice of "Authentication unsuccessful" is received by an event notice message, a random number transmission message shall be waited for the retry of authentication.

When the received message contents differ from an expected value, the message concerned shall be cancelled and a new message shall be waited. Further, if a time-out of the CTO occurs during an authentication process, all the process contents shall be cancelled and the communication shall be terminated.

NOTE The matter whether or not the communication connection retry will be executed when a communication cutoff occurred owing to a failed authentication of base station is not specified because it is a matter of practical installation requirement.

3.2.3.5 Management Service Process

3.2.3.5.1 Management Service Process Procedure Element

3.2.3.5.1.1 Access Control Information Format

In the management service process, the control information to specify the management service procedure is contained in the access control information, and the message owned by a base station and a mobile station commonly is defined.

3.2.3.5.1.1.1 Event Notify Message

This message is a message that shall be notified to the peer entity when an event such as error, etc. occurred in the ASL-ELCP is recognized.

The format of "event notify message" shall be shown in Table 3.2-18.

Table 3.2-18 — Event Notify Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier linkControlManagement (0)				Protocol identifier eventReport (3)			
2	Option Index	Status identifier “status”						
3	extensionParameter length							
	extensionParameter contents							

(1) Access point identifier
This identifier shall contain an identifier “linkControlManagement” “0” to indicate the communication control management.

(2) Protocol identifier
This identifier shall contain an identifier “evenReport” “3”.

(3) Option field of protocol identifier
(a) Option identifier (Optional Index)
This identifier identifies whether or not event additional information (extensionParameter) is added.

(b) Status identifier (status)
This identifier indicates occurred event contents. Refer to Table 3.2-19 for details.

(c) Length identifier of event additional information
This identifier shall indicate the data length of succeeding event additional information.

The unit is octet.

The area size of this length identifier shall be expanded complying with ASN.1 encoding rule.

(d) Event additional information contents
The Event additional information contents contain an inconstant length data.

Table 3.2-19 — Status Identifier (status) Contents

Value	Meaning	Notice destination	Contents of ExtensionParameter
0	Disabled to use		None
1	No access points existed	The other party station	Option
2	Function unsupported	Own station/ The other party station	Option
3	Sub-protocol unsupported	The other party station	Option
4	Data size exceeded the upper limit	Own station	Contains all parameters delivered by Service request primitive (See NOTE below.)
5	No vacancy in transmission queue; Required service cancelled	Own station	Contains all parameters delivered by Service request primitive (See NOTE below.)
6	Specified multicast link address isn't valid	Own station	Contains all parameters delivered by Service request primitive (See NOTE below.)
7	Not corresponds to the specified version.	Base station	None
8-93	Reserved for a future use		None
94	Access allowed	Mobile station	None
95	Access denied	Mobile station	None
96	Communication connection notice	Own station	Contains UserProfile type parameter
97	Communication connection notice	Own station	Contains UserProfile type parameter
98-127	Reserved for a future use		

NOTE In practical service applications, it is assumed that appropriate countermeasures will be provided such as that "the service primitive issuing origin (in own station)" holds data transmissions by returning the primitive to the service primitive issuing origin. Further, the ASL-NCP side treatments to correspond to such countermeasures comply with the specifications in each ASL-NCP.

3.2.3.5.1.1.2 Echo Transmission Message

This message is for a data transmission of the loop back communication using the ASL-ELCP as a loop back point. The table 3.2-20 shows the echo transmission message format.

Table 3.2-20 — Echo Transmission Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier linkControlManagement (0)				Protocol identifier echo (1)			
2	Dummy				Transmission origin access point identifier			
3	length for Echo data							
	contents for Echo data							

(1) Access point identifier
This identifier contains an identifier “linkControlManagement” (0) to indicate the communication control management.

(2) Protocol identifier
This identifier contains an identifier “echo” (1) to indicate the echo transmission message.

(3) Option field of protocol identifier
This identifier contains the following contents as a result of the encoding of “MsEchoParameter” type.

(a) Boundary adjustment data (dummy)
This data is a dummy data to match the data arrangement with the octet boundary.

The value shall be specified that all bits are zero.

(b) Transmission origin access point identifier
This identifier identifies an ASL-NCP.

The ASL-NCP identifier of the echo transmission requested side is contained.

(c) Echo data length identifier
This identifier indicates the data length of succeeding echo data. The unit is octet.

If no additional echo data exists, the value of the length identifier shall be zero.

The area size of this length identifier is expanded complying with ASN.1 encoding rule.

(d) Echo data contents
The echo data contents contain an undefined length data.

3.2.3.5.1.1.3 Echo Response Message

This message is a message to transmit a response data of a loop back communication by using the ASL-ELCP as a loop back point. The table 3.2-21 shows the echo response message format.

Table 3.2-21 — Echo Response Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier linkControlManagement (0)				Protocol identifier echoReply (2)			
2	Dummy				Transmission origin access point identifier			
3	length for Echo data							

	contents for Echo data

(1) Access point identifier
This identifier contains an identifier “linkControlManagement” (0) to indicate the communication control management.

(2) Protocol identifier
This identifier contains an identifier “Replyecho” (2) to indicate the echo response message.

(3) Option field of protocol identifier
This identifier contains the following contents as a result of the encoding of MsEchoParameter type.

(a) Boundary adjustment data (dummy)
This data is a dummy data to match the data arrangement with the octet boundary.

The value shall be specified that all bits are zero.

(b) Transmission origin access point identifier
This identifier identifies the ASL-NCP.

The ASL-NCP identifier of the echo transmission requested side is contained.

(c) Echo data length identifier
This identifier indicates the data length of succeeding echo data. The unit is octet.

If no additional echo data exists, the value of the length identifier shall be zero.

The area size of this length identifier shall be expanded complying with ASN.1 encoding rule.

(d) Echo data contents
The echo data contents contain an undefined length data according to the echo data contents delivered by the echo transmission message.

3.2.3.5.2 Management Service Processing Procedure

3.2.3.5.2.1 Event Process

The communication control management shall perform an event process to notify event contents and the status to its own station or the other peer station when an event such as error, etc, occurred in the ASL-ELCP.

When the event contents and status are notified to the peer entity, the “event notify” message shall be used.

When event contents and the status are notified to its own entity, event contents and the

status shall be notified complying with the specification of the event notify primitive (EventInfromation.indication).

3.2.3.5.2.2 Echo Process

The echo process provides the loop back communication function to the ASL-NCP by using the communication control management of the other peer station as a loop back point.

(1) Echo process of transmission origin
When the ASL-NCP calls an echo transmission request primitive (Echo.request), the communication control management shall generate an echo transmission message and transmit the message to the communication control management of the other party station.

When the communication control management receives an echo response message, the communication control management shall identify the ASL-NCP of the notice destination by the transmission origin access point identifier contained in the message, and the communication control management shall notify the arrival of the response to the other peer station by an echo response notice primitive (EchoReply.indication).

If an echo data is contained in the echo transmission message at that time, the contents shall be contained in the parameter “echo parameter” of the echo response notice primitive.

(2) Echo process at transmission destination
When the communication control management of the other party station receives an echo transmission message, an echo response message shall be generated and responded according to the option field contents of the echo transmission message.

3.2.3.5.2.3 MIB Access Processing Procedure

The contents concerning MIB access are a matter of practical application specifications; therefore such matters are not specified here.

3.3 Network Control Protocol (ASL-NCP)

3.3.1 Overview

3.3.1.1 Function

The ASL-NCP consists of control protocol groups, which are compliant with each protocol connected in order to encapsulate multi protocols to support multi applications.

Each of the ASL-NCP control protocol group has following functions, and provides interfaces with upper layer protocols.

(1) Passing through of upper layer protocol data unit
(2) Initial setting to use upper layer protocols
(3) Management control according to upper layer protocol nature (optional)

3.3.1.2 ASL-NCP Structure

Figure 3.3-1 shows the structure of an ASL-NCP. The basic functions of the ASL-NCP are the data transfer process for upper layer protocol encapsulation and the initial setting as the ASL-NCP management. The management control function can be incorporated into the ASL-NCP management as necessary.

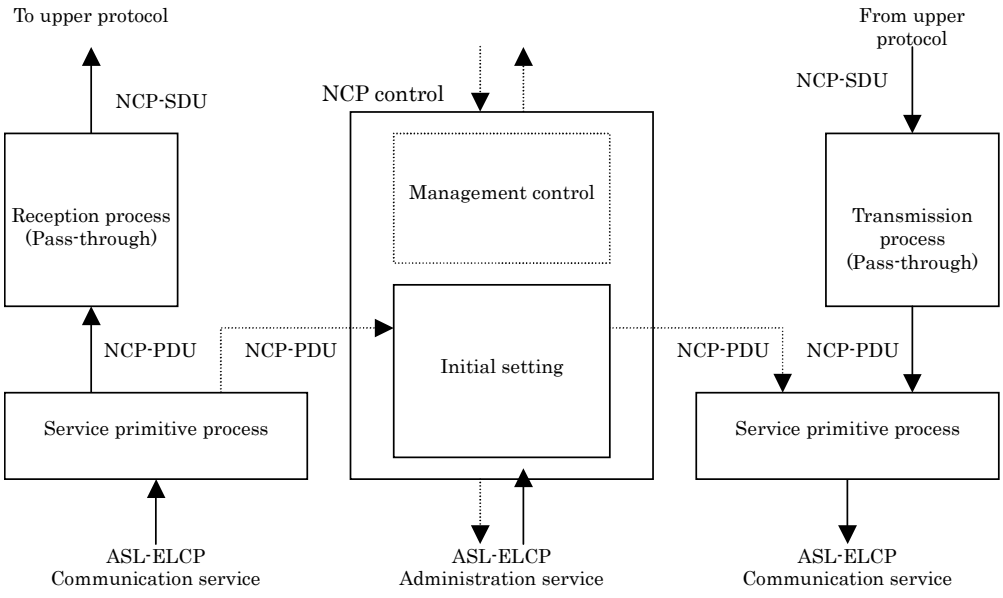


Figure 3.3-1 — ASL-NCP Structure

3.3.1.3 Service Interface

An ASL-NCP defines interoperations with upper layer protocols only. Service specifications are not defined in principle.

NOTE Service specifications are defined for an LPCP, which works as an interface with non-network applications.

3.3.1.4 Protocol

An ASL-NCP procedure is defined in the access control information added by the ASL-NCP.

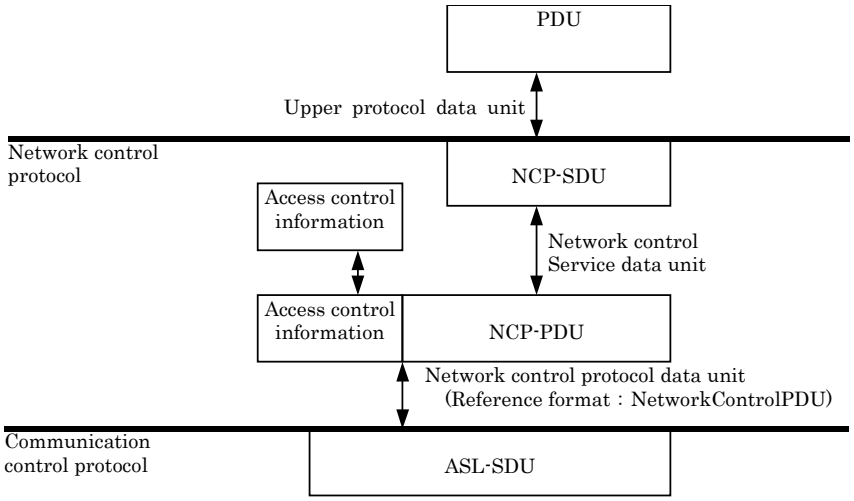


Figure 3.3-2 — Data Unit Relation between Protocols

3.3.2 ASL-NCP Common Specification

3.3.2.1 Protocol Data Unit (PDU)

3.3.2.1.1 PDU Format

NCP-PDU consists of control field (access control information), which stores control information that directs ASL-NCP procedure, and information field, which stores NCP-SDU passed by upper layer protocols, as shown in Figure 3.3-3.

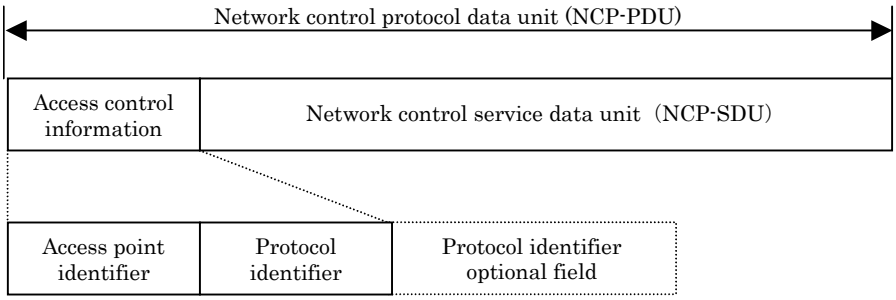


Figure 3.3-3 — PDU Structures of the ASL-NCP

3.3.2.1.2 PDU Elements

3.3.2.1.2.1 Connection Identification

The access point identifier for the ASL-NCP identification is sent or received in the NCP-PDU control field. In addition, link addresses, etc. for connection identification for each ASL-NCP is sent/received as a service primitive parameter provided by the ASL-ELCP.

3.3.3.1.2.2 Control Field Format

The control field, which directs the ASL-NCP procedure, contains access control information that consists of an access point identifier sub field and a protocol identifier sub field.

Table 3.3-1 shows the access control information field format. This access control information is defined by the "aslAccessProtocol" primitive, which is a "NetworkControlPDU" format parameter. It defines the NCP-PDU configuration of the ASL-NCP. (Refer to Annex B.)

Table 3.3-1 — Access Control Information Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier				Protocol identifier			
2	Optional field of protocol identifier							

(1) Access point identifier sub field
Values for the ASL-NCP identification shown in Table 3.3-2 are stored in the access point identifier field.

Table 3.3-2 — Access Point Identifier

Access point identifier	Control protocol name	Remarks
0	Communication control administration	Protocol process in the communication control management
1	Local port control protocol	Access point to local environment
2	LAN control protocol	Access point to LAN environment
3	PPP control protocol	Access point to PPP environment
4-13	Reserved	
14	Local port control protocol 2	Access point to local environment (for road operators)
15	Prohibited	

(2) Protocol identifier sub field
Protocol identifier sub field stores control information that directs the ASL-NCP procedure. Furthermore, this sub field has an optional field that multiplexes additional information to supplement control contents. It is utilized as necessary when protocols of the same layer are formed between the same ASL-NCP entities.

The contents of this protocol identifier sub field are defined separately for each ASL-NCP.

3.3.2.1.2.3 Information Field Format

When data is transferred, it is stored in information field after passed through the NCP-SDU passed by upper layer protocols.

The information field other than the above shall be NULL (data length is “0”.).

3.3.3 Local Port Control Protocol (LPCP)

3.3.3.1 Overview

The LPCP is a control protocol that provides the data transfer service to the upper layer protocol such as applications and the management service to provide communication means to non-network type applications whose model type typically is the client/server type or peer-to-peer type.

3.3.3.2 Local Port

In order to perform multiple non-network type applications, the LPCP defines the identification information (local port) to identify connection for the upper layer protocol.

In order to send data correctly from a sending source application to a destination application, the LPCP identifies connection of each application using local ports that identify the sending destination application and sending source application and a link address that identifies the counterpart station.

NOTE With regard to numbers (local port numbers) given to local ports for identification of applications, there are two types of port No.—"reserved port No." and "private port No". For the details of local port number assignment, refer to Annex H.

3.3.3.3 LPCP Interface Service Specification

3.3.3.3.1 Interaction Overview

3.3.3.3.1.1 Data Transfer Service Interface

The LPCP provides the following primitives as the data transfer service to the upper layer protocol.

TransferData.request
TransferData.indication

The "TransferData.request" is passed from the upper layer protocol to the LPCP to request to transfer the NCP-SDU passed from the upper layer protocol to the remote station.

The "TransferData.indication" is passed from the LPCP to the upper layer protocol to indicate arrival of the NCP-SDU.

3.3.3.3.1.2 Management Service Interface

The LPCP provides the following management services to the upper layer protocol.

(1) Event notify service

In the event notify service, the LPCP provides the following primitive to the upper layer protocol.

EventReport.indication

The “EventReport.indication” is passed from the LPCP to the upper layer protocol in the local station to notify of an event notified by the event notification service of the ASL-ELCP, or passed from the LPCP to the upper layer protocol in the remote station or local station to provide notification that an event such as error occurred in the LPCP.

(2) Local port management services
In the local port management service, the LPCP provides the following primitives to the upper layer protocol.

OpenPort.request
OpenPort.confirm
ClosePort.request

The “OpenPort.request” is passed from the upper layer protocol to the LPCP to request the opening of a local port. The “OpenPort.confirm” is passed from the LPCP to the upper layer protocol to provide notification of the opened local port number. The “ClosePort.request” is passed from the upper layer protocol to the LPCP to request the closing of a local port.

3.3.3.3.2 Service Content Specification

This subclause specifies primitives and parameters related to the data transfer service and management service. Parameters are described abstractly as interfaces, and the information required for receiver entity are specified. A specific implementation is not constrained in the method of making this information available.

3.3.3.3.2.1 Data Transfer Service Interface

The parameter “destinationPort” indicates the local port number (application), which is the data sending destination, and together with the parameter “linkAddress” identifies the access point of the local port control protocol.

The parameter “sourcePort” indicates the local port number (application) that is the data sending source. When a response from the sending destination is required, it is used as the default access point for giving the response.
NOTE When any access point (local port number) other than the default access point is used, the application specifies it respectively. For details, refer to the Annex H.

The parameter “userData” is provided by the actual NCP-SDU itself or by passing a pointer to the NCP-SDU or by other means.

The LPCP handles these parameters from the upper layer protocol in the following specification.

Table 3.3-3 — Data Transfer Service Parameters

Parameter name	ASN.1 type	Remarks
linkAddress	DsrcLID	
sourcePort	PortNo	
destinationPort	PortNo	
userData	OCTET STRING	

Figure 3.3-4 shows the logical relationship among primitives in the data transfer service provided by the LPCP to the upper layer protocol.

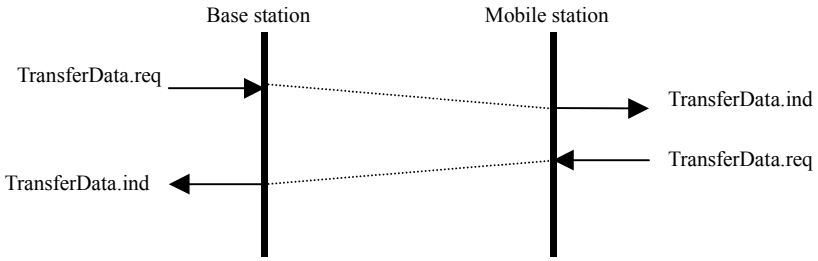


Figure 3.3-4 — Logical Relationship between Data Transfer Service Primitives

3.3.3.3.2.2 Management Service Interface

The parameter “destinationPort” indicates the local port number (application) that is the data sending destination, and identifies the access point of the LPCP together with the parameter “linkAddress”.

The parameter “eventCode” indicates the type of event that occurred.

The parameter “extensionParameter” indicates the additional event information corresponding to each event code, and may be provided by the actual data itself or by passing pointer to the data or by other means.

The parameter “openPort” indicates the number of an opened local port.

The parameter “primitiveType” indicates the type of the indication primitive received by the opened local port.

The parameter “recvEventCode” indicates the type of event received by the opened local port.

The parameter “closePort” sets the local port number to be closed.

The LPCP handles these parameters passed from the upper layer protocol in the following specification.

Table 3.3-4 — Management Service Parameters

Parameter name	ASN.1 type	Remarks
linkAddress	DsrcLID	
destinationPort	PortNo	
eventCode	LpcpEventCode	
extensionParameter	OCTET STRING	
openPort	INTEGER (0..65535)	
primitiveType	LpcpPrimitiveType	
recvEventCode	LpcpEventCode	
closePort	INTEGER (0..65535)	

Figure 3.3-5 shows the logical relationship among primitives in the management service provided by the LPCP to the upper layer protocol.

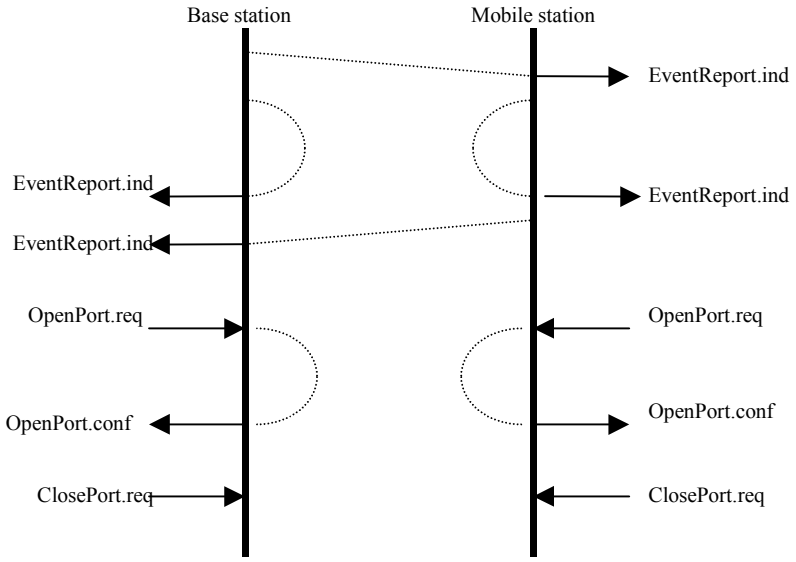


Figure 3.3-5 — Logical Relationship between Management Service Primitives

3.3.3.3.3 Data Transfer Service

3.3.3.3.3.1 Data Transfer Request Service Primitive (TransferData.request)

(1) Function

This primitive is to request transfer of the NCP-SDU to the remote station.

(2) When generated
This primitive is generated by the upper layer protocol.

(3) Semantics of service primitive
This primitive shall provide parameters as follows.

TransferData.request (linkAddress, sourcePort, destinationPort, userData)

The parameter “linkAddress” is the link address used in the DSRC. The “linkAddress” parameter in the mobile station is a private link address. The parameter “linkAddress” in the base station is a private link address or multicast link address. When the parameter “linkAddress” is a multicast link address, the NCP-SDU is delivered in the broadcast mode.

The parameter “sourcePort” indicates the local port number of the sending source.
The parameter “destinationPort” indicates the local port number of the sending destination.

The parameter “userData” indicates the NCP-SDU passed from the upper layer protocol in the local station.

3.3.3.3.3.2 Data Transfer Indication Service Primitive (TransferData.indication)

(1) Function
This service primitive provides notification of reception of the NCP-SDU from the remote station.

(2) When generated
This primitive is generated by the LPCP to indicate reception of the NCP-SDU.

(3) Semantics of service primitive
This primitive shall provide parameters as follows:

TransferData.indication (linkAddress, sourcePort, destinationPort, userData)

The parameter “linkAddress” is the link address used in the DSRC. The “linkAddress” parameter in the mobile station is a private link address. The parameter “linkAddress” in the base station is a private link address or multicast link address.

The parameter “sourcePort” indicates the local port number of the sending source.
The parameter “destinationPort” indicates the local port number of the sending destination.

The parameter “userData” indicates the received NCP-SDU.

3.3.3.3.4 Management Services

3.3.3.3.4.1 Event Notify Service

This service provides notification of an event notified by the event notification service of the ASL-ELCP or provides notification that an event such as an error occurred in the LPCP. The LPCP provides the following service primitive.

3.3.3.3.4.1.1 Event Notify Indication Service (EventReport.indication)

(1) Function
This primitive provides notification of an event from the event notification service of the ASL-ELCP or provides notification that an event such as an error occurred in the LPCP.

(2) When generated
This primitive is generated by the LPCP when notification of an event such as an error is provided by the event notification service of the ASL-ELCP or when an event such as an error occurred in the LPCP is provided.

(3) Semantics of service primitive
This primitive shall provide parameters as follows:

EventReport.indication (linkAddress, destinationPort, eventCode, [extensionParameter])

The parameter “linkAddress” indicates the link address used in the DSRC.
The parameter “destinationPort” indicates the local port number of the application that will be notified of the event.

The parameter “eventCode” indicates the code indicating the event that occurred.

The parameter “extensionParameter” indicates the information to supplement the contents of the parameter “eventCode” when necessary. This parameter is omitted for NULL data (data whose length is “0”).

3.3.3.3.4.2 Local Port Management Service

This service opens or closes local ports for transferred data and occurred events. The LPCP provides the following service primitives.

3.3.3.3.4.2.1 Local Port Open Request Service Primitive (OpenPort.request)

(1) Function
This service primitive is to request the opening of a local port for receiving data or events.

(2) When generated

This primitive is generated by the upper layer protocol.

(3) Semantics of service primitive
This primitive shall provide parameters as follows:

OpenPort.request ([openPort], [primitiveType], [recvEventCode])

The parameter “openPort” is indicates the local port number to be opened. This parameter is optional. When this parameter is omitted, LPCP assigns a local port number.

The parameter “primitiveType” indicates the identifier to specify the indication primitive type received by the opened local port. This parameter is optional. When this parameter is omitted, the request is to receive all indication primitives. For details, refer to Table 3.3-5.

The parameter “recvEventCode” indicates the identifier to specify the event type received by the opened local port. This parameter is optional. When this parameter is omitted, the request is to receive all events. The content is equivalent to those of the event code (eventCode). (Refer to Table 3.3-9)

NOTE When the upper layer protocol uses LPCP; it is required to give the request for communication connection notice by using the local port open primitive in advance.

Table 3.3-5 — Primitive Type Identifier (primitiveType) Description

Primitive type	Description	Remarks
0	All primitives	
1	Data transfer indication primitive	
2	Event notification primitive	

3.3.3.3.4.2.2 Local Port Open Confirmation Service Primitive (OpenPort.confirm)

(1) Function
This service primitive provides notification of the local port number opened to receive data or events.

(2) When generated
This primitive is generated by the LPCP when the OpenPort.request primitive is issued.

(3) Semantics of service primitive
This primitive shall provide parameters as follows:

OpenPort.confirm ([openPort])

The parameter “openPort” indicates the opened local port number. This parameter is optional. When this parameter is omitted, it provides notification that opening of the specified local port has failed.

3.3.3.3.4.2.3 Local Port Close Request Service Primitive (ClosePort.request)

(1) Function
This service primitive is to request the closing of an opened local port.

(2) When generated
This primitive is generated by the upper layer protocol.

(3) Semantics of service primitive
This primitive shall provide parameters as follows:

ClosePort.request (closePort)

The parameter “closePort” indicates the local port number to be closed.

3.3.3.4 LPCP Procedure Element

3.3.3.4.1 Maximum Transfer Unit (MTU) of LPCP

The MTU is the maximum length of data, which can be passed from the LPCP to the ASL-ELCP. The MTU of the LPCP shall be “522” octets (including the access control information).

3.3.3.4.2 Access Control Information Format

LPCP shall store a protocol identifier of LPCP and an option field type as shown in Table 3.3-6 in the access control information to define the messages shared by the base station and the mobile station.

Table 3.3-6 — Protocol Identifier of LPCP

Protocol identifier	Description	Option field type (Refer to Annex B.)
0	Event notification (eventReport)	“LpcpEventParameter” type
1	Data transfer (message)	“LpcpTransferDataPDU” type
2-15	Reserved	

3.3.3.4.2.1 Data Transfer Message

This message is provided to transfer the PDU of the upper layer protocol. Table 3.3-7 shows the data transfer message format.

Table 3.3-7 — Data Transfer Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier “localPortControl(1)” or “localPortControl2(14)”				Protocol identifier “message(1)”			
2	Source local port number (high-order)							
3	Source local port number (low-order)							
4	Destination local port number (high-order)							
5	Destination local port number (low-order)							

(1) Access point identifier
This field shall set the identifier “localPortControl (1)” or “localPortControl2 (14)” indicating the LPCP.

(2) Protocol identifier
This field shall set the identifier “message (1)” indicating the data transfer message.

(3) Optional fields
This field shall set the following contents as the result of the “LpcpTransferDataPDU” type coding.

(a) Source local port number
This field shall set the local port number of the sending source.

(b) Destination local port number
This field shall set the local port number of the sending destination.

3.3.3.4.2.2 Event Notify Message

This message is provided to notify of an event by the event notification service of the ASL-ELCP or an event occurred in the LPCP. Table 3.3-8 shows the event notification message format.

Table 3.3-8 — Event Notify Message Format

	7(MSB)	6	5	4	3	2	1	0(LSB)
1	Access point identifier “localPortControl(1)” or “localPortControl2(14)”				Protocol identifier “eventReport(0)”			
2	Event code “eventCode”							
3	Length of “extensionParameter”							
	Contents of “extensionParameter”							

(1) Access point identifier
This field shall set the identifier “localPortControl (1)” or “localPortControl2 (14)” indicating the LPCP.

(2) Protocol identifier
This field shall set the identifier “eventReport (0)” indicating the event notification message.

(3) Optional fields
This field shall set the following contents as the result of the “LpcpEventParameter” type coding.

(a) Event code
This identifier shall specify the details of the occurred event. The codes from “0” to “127” identify the ASL-ELCP status. The codes from “128” to “255” identify the LPCP status. For details, refer to Table 3.3-9.

(b) Length of extensionParameter
This field shall set the data length of the following “extensionParameter”. The unit is octet. The size of this field expands according to the ASN.1 encoding rule. If no event information follows (that is, in the case of null), “0” is set in this field.

(c) Contents of extensionParameter
This field shall set the contents of “extensionParameter”.

Table 3.3-9 — EventCode Descriptions

eventCode	Description	Notification target	Contents of “extensionParameter”
0	Prohibited to use.		None
1-3	Not use.		None
4	The data size exceeds the upper limit value.	Local (own) station	
5	The sending service is aborted because of sending queue overflow.	Local (own) station	
6	The multicast link address is invalid.	Local (own) station	
7-93	Reserved For Future Use		None
94-95	Not use.		None
96	Connection notice	Local (own) station	The “UserProfile” type parameter is stored.
97	Disconnection notice	Local (own) station	The “UserProfile” type parameter is stored.
98-127	Reserved For Future Use		
128	The DSRC is not connected.	Local (own) station	
129	The destination port is invalid.	Remote (peer) station	The “InvalidPort” type parameter is stored.
130	Accept port list	Remote (peer) station	The “PortList” type parameter is stored.
131-255	Reserved For Future Use		

3.3.3.4.3 Control Information of LPCP

3.3.3.4.3.1 Accept Local Port List

The “accept (able) local port list” consists of the local port No. (openPort), notification primitive type (primitiveType) and notification event type (recvEventCode) passed from OpenPort.request, and is used to identify whether or not the destination port of receiving data is opened.

The LPCP adds a local port to the list when it receives the “OpenPort.request”, and deletes a local port from the list when it receives the “ClosePort.request”.

3.3.3.4.3.2 Communication Control Information List

The “communication control information list” consists of the link address and the “UserProfile” type parameter passed from the status “connection notice” through the “EventInformation.indication” from ASL-ELCP, and is used to indicate whether or not the DSRC is connected when the “TransferData.request” is received.

The LPCP adds the information to the list when it receives the status “connection notice” from the ASL-ELCP through the “EventInformation.indication” in the management service, and deletes the information from the list when it receives the status “disconnection notice” through the “EventInformation.indication” from ASL-ELCP.

3.3.3.5 LPCP Procedure

3.3.3.5.1 Local Port Management Procedure

3.3.3.5.1.1 Local Port Open Processing

When the “OpenPort.request” in which the parameter “openPort” is specified is received from the upper layer protocol, the LPCP confirms the “accept local port list”. If the specified local port (openPort) does not overlap an existing port, the LPCP registers the local port Number. (openPort), notification primitive type (primitiveType) and notification event type (recvEventCode) in the accept local port list, and then notifies the upper layer protocol of the opened local port number through the “OpenPort.confirm”. If the specified local port (openPort) does overlap an existing port, the LPCP does not register the contents above, and notifies the upper layer protocol that opening of the local port has failed through the !OpenPort.confirm” with the “openPort” parameter omitted.

When receiving an “OpenPort.request” in which the parameter “openPort” is omitted, the local port control protocol shall assign a local port number to the requested process, register the assignment result, notification primitive type (primitiveType) and notification event type (recvEventCode) in the accept local port list, and then notify the upper layer protocol of the opened local port number through the “OpenPort.confirm”. At the time of number

assignment, the local port control protocol shall check the “accept local port list”, and assign a non-overlapping number.

3.3.3.5.1.2 Local Port Close Processing

When receiving the “ClosePort.request” from the upper layer protocol, the LPCP deletes the information on the requested local port number from the “accept local port list”, and will not perform the notification of the received message after that for the deleted local port.

3.3.3.5.2 Connection Processing Procedure

3.3.3.5.2.1 Communication Connection Process

When receiving the “connection notice” through the “EventInformation.indication” in the management service of the ASL-ELCP, the LPCP registers the received link address and “UserProfile” type parameter in the communication control information list.

And then the LPCP notifies the local port of a “connection notice” through the “EventReport.indication”. This notification procedure is applied to a local port whose contents in the “accept local port list” are either of the following.

(1) When the content of “primitiveType” is omitted or “all primitives (0)”.
(2) When the content of “primitiveType” is “event notification service (2)”, and “recvEventCode” is omitted.
(3) When the content of “primitiveType” is “event notification service (2)”, and the content of “recvEventCode” is “connection notice (96)”.

After that, the LPCP refers to the accept local port list, generates the event notification message whose event code is “accept port list (130)” and whose additional event information is the accept local port list in the local station, and then sends the message to the remote (peer) station.

When receiving the event notification message (“accept port list (130)”) sent from the remote (peer) station, the LPCP notifies the local port of the event “accept port list (130)” through the “EventReport.indication”. This notification procedure is applied to a local port whose contents in the “accept local port list” are either of the following.

(1) When the content of “primitiveType” is omitted or “all primitives (0)”.
(2) When the content of “primitiveType” is “event notification service (2)”, and “recvEventCode” is omitted.
(3) When the content of “primitiveType” is “event notification service (2)”, and the content of “recvEventCode” is “accept port list (130)”.

Figure 3.3-6 shows the outline of the Initial set up procedure of the LPCP according to the procedure above.

This procedure is performed only in the point-to-point type communication.

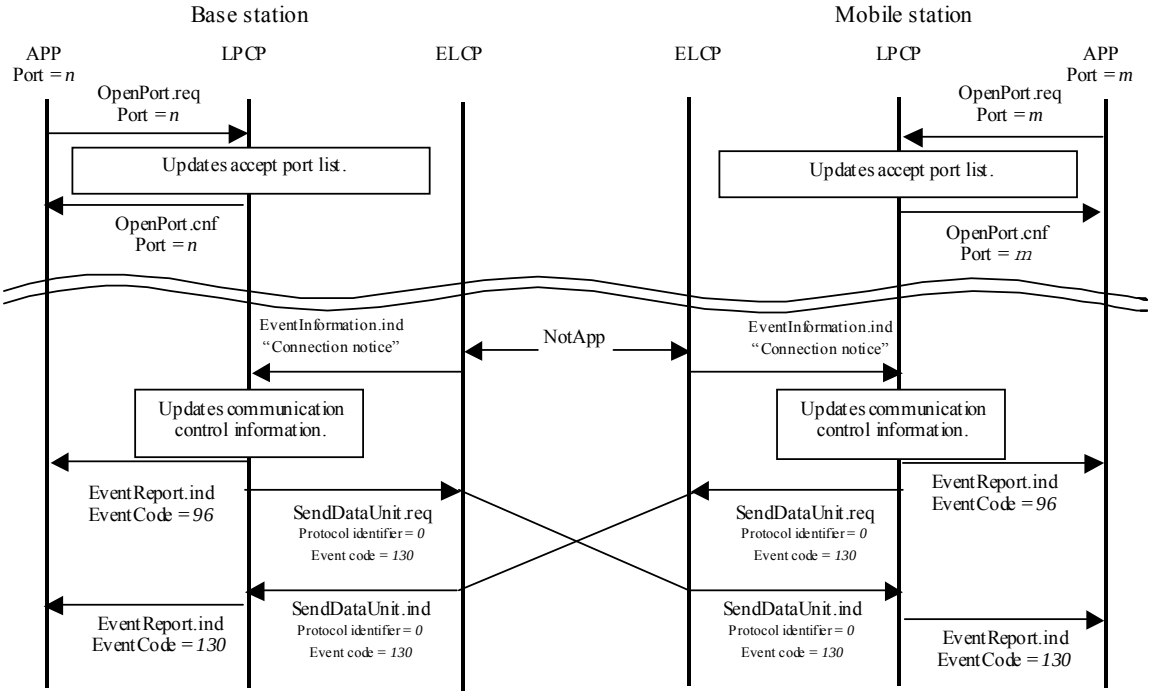


Figure 3.3-6 — Communication Connection Process Outline

3.3.3.5.2.2 Communication End Process

When receiving "disconnection notice" through the EventInformation.indication in the management service of the ASL-ELCP, the LPCP deletes the information on the received link address from the communication control information list.
And then the LPCP notifies the local port of a "disconnection notice" through the EventReport.indication. This notification procedure is applied to a local port whose contents in the accept local port list are either of the following.

(1) When the content of "primitiveType" is omitted or "all primitives (0)".
(2) When the content of "primitiveType" is "event notification service (2)", and "recvEventCode" is omitted.
(3) When the content of "primitiveType" is "event notification service (2)", and the content of "recvEventCode" is "communication disconnection notice (97)".

Figure 3.3-7 shows the outline of the communication end procedure of the LPCP according to the procedure above.

This procedure is performed only in the point-to-point type communication.

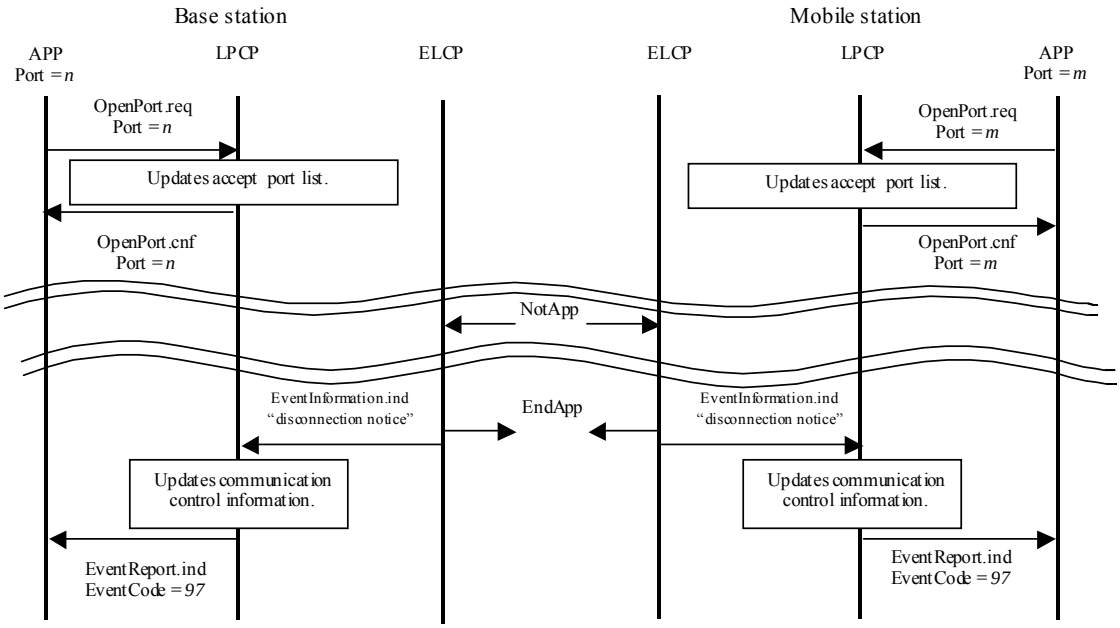


Figure 3.3-7—Communication End Process Outline

3.3.3.5.3 Data Transfer Procedure

Figure 3.3-8 shows the outline of the data transfer procedure.

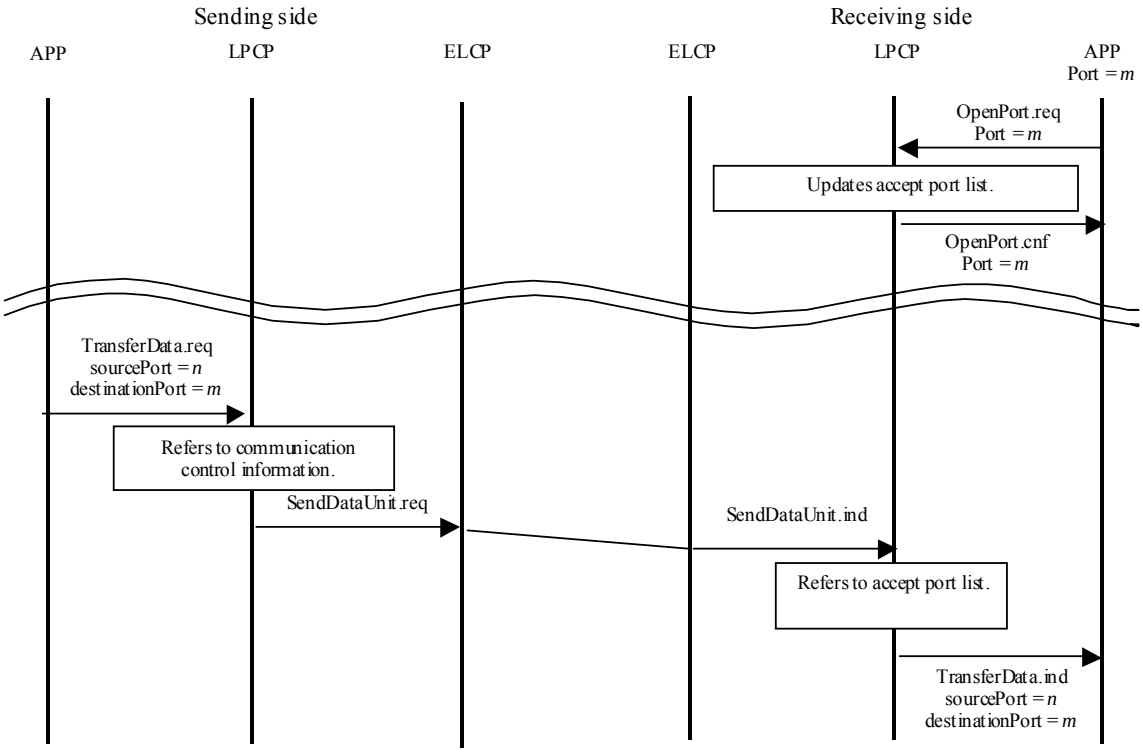


Figure 3.3-8 — Data Sending Process Procedure Outline

3.3.3.5.3.1 Data Sending Process

When receiving the "TransferData.request" primitive from the upper layer protocol, the LPCP refers to the communication control information list, and then sends the NCP-SDU passed from the primitive through the data transfer message if the link address of the primitive is valid.

At this time, the contents passed from the primitive shall be set in the sending source local port number and sending destination local port number in the data transfer message.

When receiving the "TransferData.request" from the upper layer protocol, if the specified link address is a private one and the DSRC is not connected, the LPCP sends the source local port number specified by the "TransferData.request" a "DSRC not connected" through the "EventReport.indication".

However, this event notification procedure is performed only when the event notification request was passed from the "OpenPort.request" primitive. If the event notification request was not given, the LPCP does not notify the upper layer protocol of the event. The necessity of event notification is judged based on the "accept local port list".

Figure 3.3-9 shows the outline of the processing procedure when the DSRC is not connected.

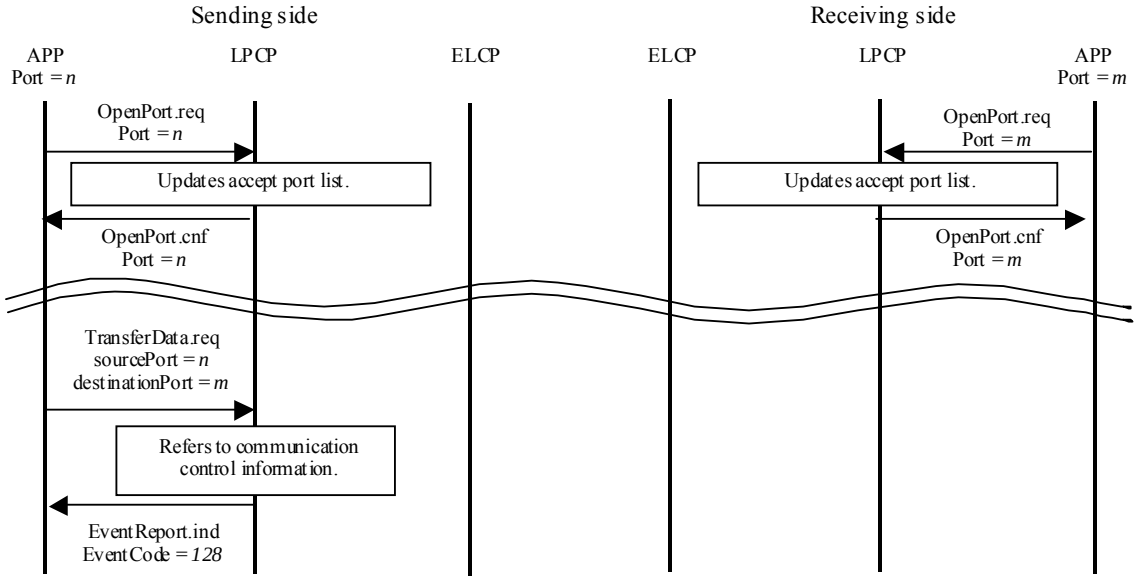


Figure 3.3-9 — Outline of Processing Procedure when DSRC is not connected

When receiving the event notification message from the remote station to which the data transfer message was sent, the LPCP confirms the event code given in the message.

When the contents of the event code are “The sending destination local port is invalid.” the LPCP notifies the source local port specified in the “extensionParameter” of a “The destination local port is invalid.” through “EventReport.indication”.

However, this event notification procedure is performed only when the event notification request was passed from the “OpenPort.request” primitive. If the event notification request was not given, the LPCP does not notify the upper layer protocol of the event. The necessity of event notification is judged based on the accepted local port list.

3.3.3.5.3.2 Data Reception Process

When receiving a data transfer message, the LPCP extracts the protocol identifier, destination local port number, source local port number and NCP-SDU from the message.

The LPCP refers to the “accept local port list”, and then notifies the upper layer protocol specified by the destination local port number that the NCP-SDU be received from the remote station through the “TransferData.indication” if the destination local port number is valid.

This data receiving processing is applied to a local port whose contents in the “accept local port list” are either of the following:

(1) When the contents of “primitiveType” are omitted or “all primitives (0)”.
(2) When the contents of “primitiveType” are “data transfer indication primitive (1)”.

When the link address is a private one and the destination local port number specified in the received data transfer message is invalid, the LPCP sends back the event notification message whose event code indicates “The destination local port is invalid.”

When the link address is a multicast one and the destination local port number specified in the received data transfer message is invalid, the LPCP aborts the received data.

Figure 3.3-10 shows the outline of the procedure when the destination local port number is invalid.

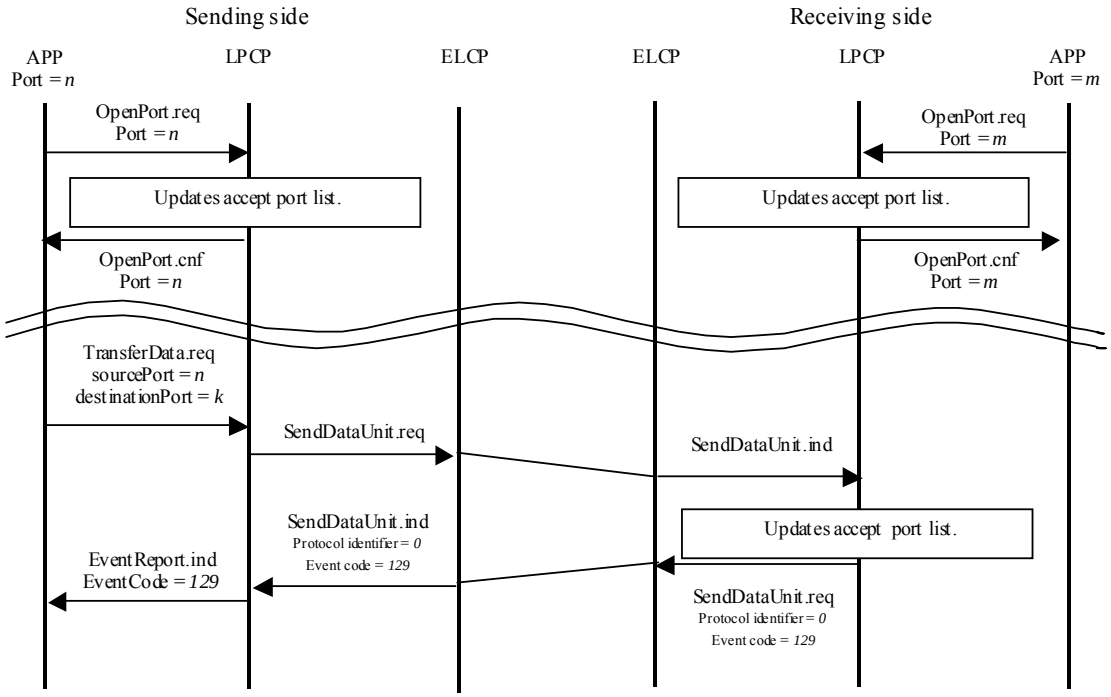


Figure 3.3-10 — Outline of Reception Procedure in case of the Invalid Destination Local Port Number

3.3.4 LAN Control Protocol (LANCP)

3.3.4.1 Overview

The LANCP is a control protocol that provides an interface with a data link layer in the LAN side in order to connect with local area network (LAN: Local Area Network) using the DSRC as a medium.

3.3.4.2 LANCP Interface

3.3.4.2.1 Network Configuration Update Setup

LANCP generates and provides a startup trigger to automatically start a network configuration setting protocol. It is based on the assumption that the mobile station uses a network configuration setting protocol such as dynamic host configuration protocol (DHCP).

For the purpose above, the mobile station acquires a server MAC address from the base station, which works as a network configuration-setting unit. After comparison between the obtained address and the address held by the mobile station, it generates the startup trigger if there is a difference between the two.

The startup trigger provision method is not specified in here. It will be based on the actual implementation.

3.3.4.2.2 Communication Frame Transfer

The LANCP stands between an ASL-ELCP and a LAN data link layer. It provides the function that pass through a communication frame exchanged by data link layers. (Refer to Figure 3.3-11.).

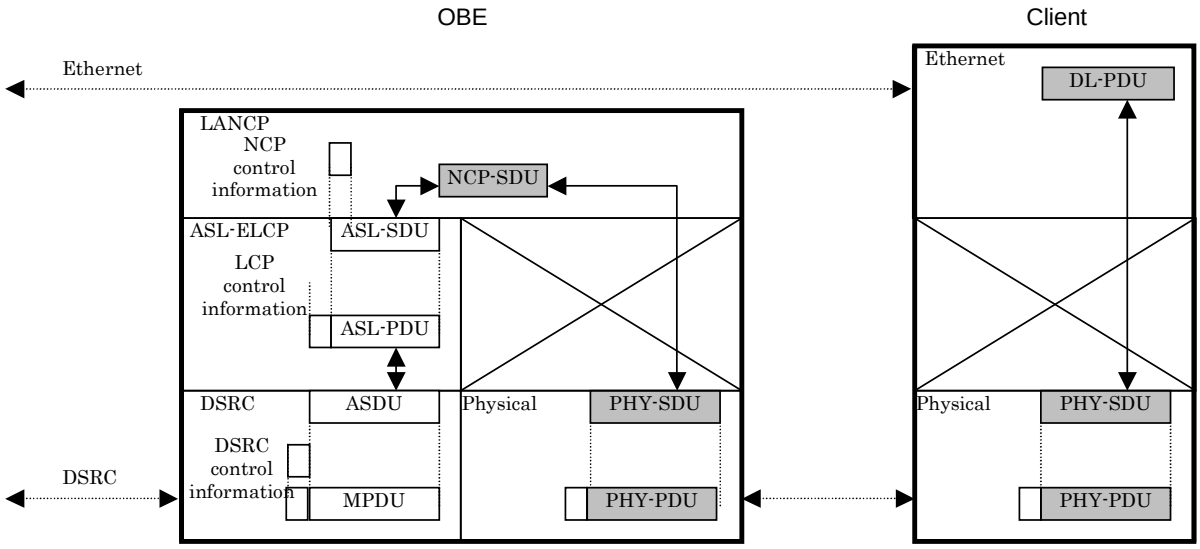


Figure 3.3-11 — Example of Communication Frame Transfer

3.3.4.3 LANCP Procedure Element

3.3.4.3.1 Upper Layer Protocol Specification

The communication frame that is the objective of the LANCP pass-through shall be IEEE 802.3 MAC sub layer data link frame. (Refer to Figure 3.3-12).

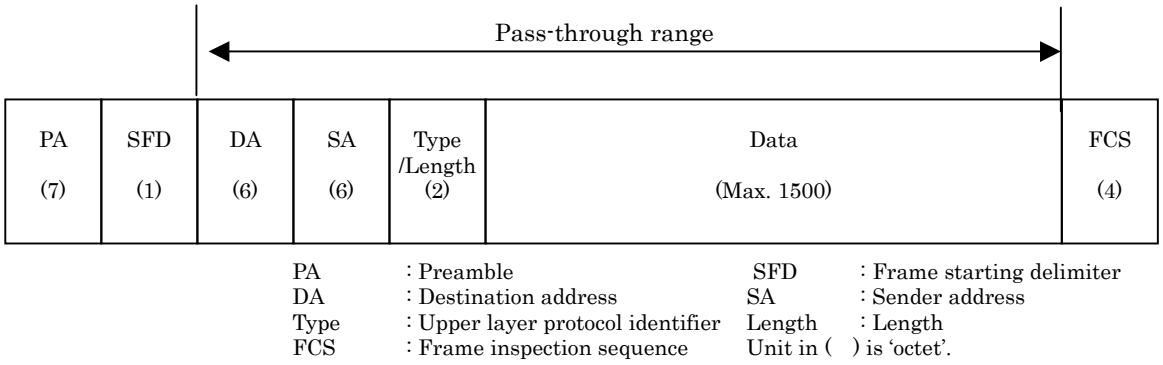


Figure 3.3-12 — Communication Frame Pass-through Range

3.3.4.3.2 LANCP Maximum Transfer Unit (MTU)

It is a maximum data unit length (MTU) that LANCP can pass to ASL-ELCP. The MTU of LANCP shall be "1517" octets (including access control information).

3.3.4.3.3 Access Control Information Format

The LANCP directs the LANCP procedure to access control information. For that purpose, it stores the control information shown in Table 3.3-10, and defines the messages shared between the base station and the mobile station.

Table 3.3-10 — LANCP Protocol Identifier

Protocol identifier	Contents	Optional field format (Refer to Annex B)
0	Data transfer	None
1	Mobile station initial setting	InitialData type
2	Base station initial setting	InitialData type
3-15	Reserved	

3.3.4.3.3.1 Data Transfer Message

This is a message for upper layer protocol PDU transfer. Data transfer message format is shown in Table 3.3-11.

Table 3.3-11 — Data Transfer Message Format

	7(MSB)	6	5	4	3	2	1	0(LSB)
1	Access point identifier lanContol (2)				Protocol identifier message (0)			

(1) Access point identifier
Identifier “lanContol (2)”, which indicates LANCP, is stored.

(2) Protocol identifier
Identifier “message (0)”, which indicates data transfer, is stored.

(3) Protocol identifier optional field
Not used.

3.3.4.3.3.2 Mobile Station Initial Setting Message

This message notifies mobile station initial setting information. Format of the mobile station initial setting message is shown in Table 3.3-12.

Table 3.3-12 — Mobile Station Initial Setting Message Format

	7(MSB)	6	5	4	3	2	1	0(LSB)
1	Access point identifier lanContol (2)				Protocol identifier obeInitialMessage(1)			
2	Client MAC address [6] (upper)							
	:							
	Client MAC address [1] (lower)							

(1) Access point identifier
The identifier “lanControl (2)”, which indicates LANCP, is stored.

(2) Protocol identifier
The identifier “obeInitialMessage (1)”, which indicates mobile station initial setting, is stored.

(3) Protocol identifier optional field
Following contents are stored as a result of the “InitialData” type encoding.

(a) Client MAC address
Client MAC address (“6” octets) is stored.

3.3.4.3.3.3 Base Station Initial Setting Message

This message notifies base station initial setting information to the mobile station. Format of the base station initial setting message is shown in Table 3.3-13.

Table 3.3-13 — Base Station Initial Setting Message Format

	7(MSB)	6	5	4	3	2	1	0(LSB)
1	Access point identifier lanContol (2)				Protocol identifier rsuInitialMessage(2)			
2	Server MAC address [6] (upper)							
	:							
	Server MAC address [1] (lower)							

(1) Access point identifier
Identifier “lanControl (2)”, which indicates LANCP, is stored.

(2) Protocol identifier
Identifier “rsuInitialMessage (2)”, which indicates mobile station initial setting, is stored.

(3) Protocol identifier optional field
Following contents are stored as a result of “InitialData” type encoding:

(a) Client MAC address
Client MAC address (“6” octets) is stored.

3.3.4.3.4 Base Station Management Table

A base station management table correlates a link address and a MAC address in order to specify a delivery destination of the NCP-SDU passed by upper layer protocols.

This management table is generated when ‘communication connection notification’ state is received in the event notification primitive “EventInformation.indication” of the administration service, and abandoned when “communication disconnection notification” state is received.

3.3.4.4 LANCP Procedure

Figure 3.3-13 shows the outline of LANCP communication procedure.

3.3.4.4.1 Connection Process Procedure

3.3.4.4.1.1 Initial Setting Process

(1) Base station initial setting process
When it receives ‘communication connection notification’ state in the event notification primitive “EventInformation.indication” of the management service, it moves to initial setting phase. Then, it awaits the initial setting information notification from the mobile station of the link address passed by the primitive.

In the case that it receives a mobile station initial setting message in the initial setting phase, it generates a management table utilizing the link address passed by the primitive and the client MAC address.

Next, it transmits a base station initial setting message in order to notify the server MAC address to the pertinent mobile station, and moves from the initial setting phase to the communication phase to implement data transfer.

If it receives “communication disconnection notification” state in the event notification primitive “EventInformation.indication” of the management service, it terminates the initial setting phase.

If it receives messages other than the mobile station initial setting message in the initial setting phase, it abandons the messages.

NOTE1 The server MAC address is assumed to be the MAC address allocated to the servers or routers, or gateways that manage the field to be the update unit of the network configuration setting.

NOTE2 The server MAC address acquisition method at the base station is not specified here. It will be based on the actual implementation.

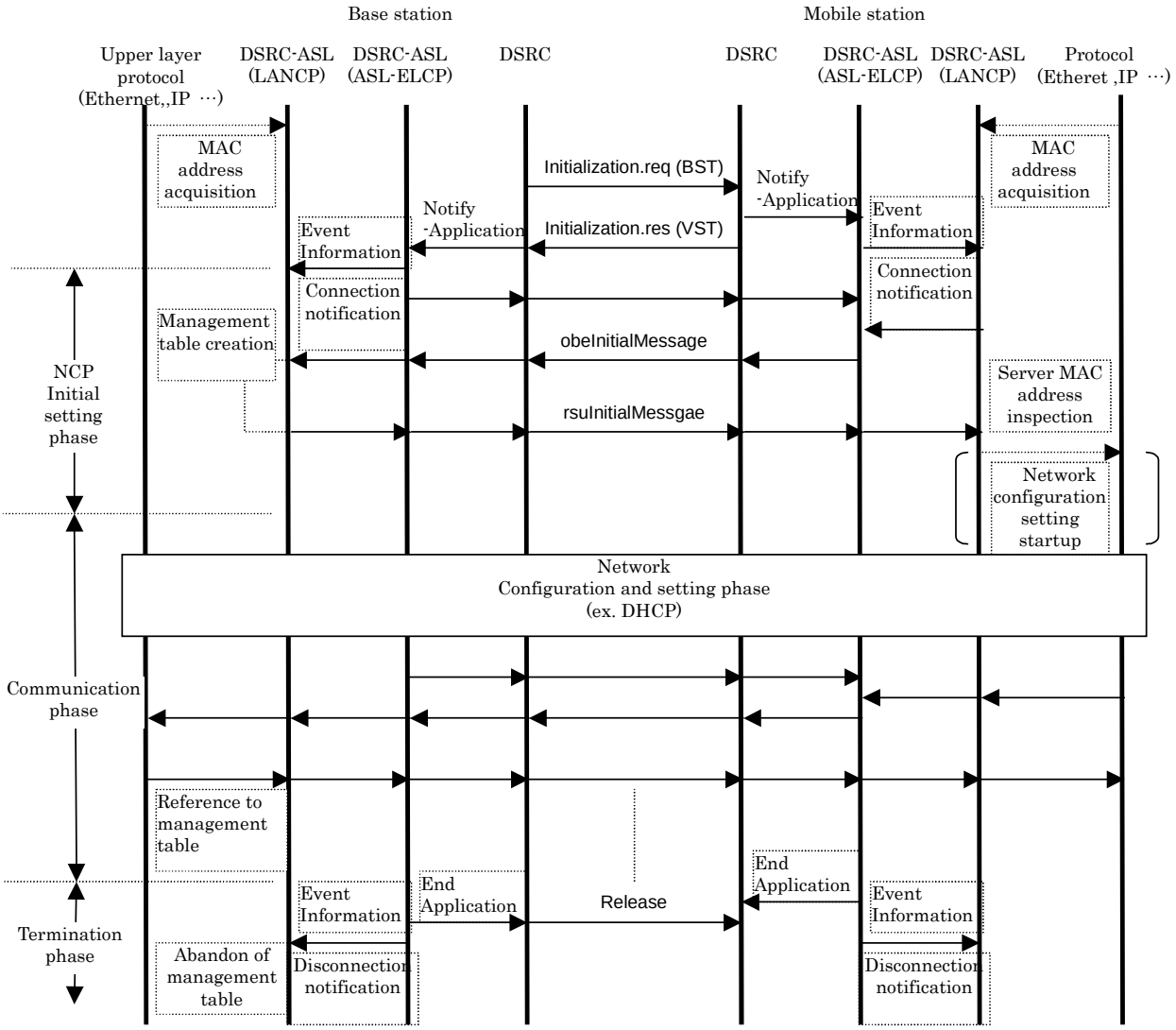


Figure 3.3-13 — LANCP Communication Procedure Outline

(2) Mobile station initial setting process

When it receives “communication connection notification” state in the event notification primitive “EventInformation.indication” of the management service, it moves to initial setting phase. Then, it transmits a mobile station initial setting message to notify a client MAC address to the base station, and awaits the response from the pertinent base station.

When it receives a base station initial setting message in the initial setting phase, it compares the server MAC address passed by the message and the MAC address held by the mobile station.

If they don’t match after the comparison, it updates the server MAC address held, as well as issues a startup trigger to prompt the network configuration setting update to upper layer

protocols. Then, it moves from the initial setting phase to the communication phase to implement data transfer.

If they match, it doesn't issue a startup trigger, but only implement transition procedure to move to the communication phase. In the case that it receives "communication disconnection notification" state in "EventInformation.indication", it terminates the initial setting phase.

If it receives messages other than the base station initial setting message in the initial setting phase, these messages shall be discarded.

NOTE1 The startup trigger provision method at mobile stations is not specified here. It will be based on the actual implementation.
NOTE2 The client MAC address acquisition method at mobile stations is not specified here. It will be based on the actual implementation.

3.3.4.4.1.2 Communication Termination Process

(1) Base station communication termination process
When it receives "communication disconnection" state in the event notification primitive "EventInformation.indication" of the administration service, it abandons the link address management table passed by the primitive and terminates the communication phase.

(2) Mobile station communication termination process
When it receives "communication disconnection" state in the event notification primitive "EventInformation.indication" of the management service, it terminates the communication phase.

3.3.4.4.2 Data Transfer Procedure

3.3.4.4.2.1 Base Station Data Transfer Process

3.3.4.4.2.1.1 Individual Communication Processing

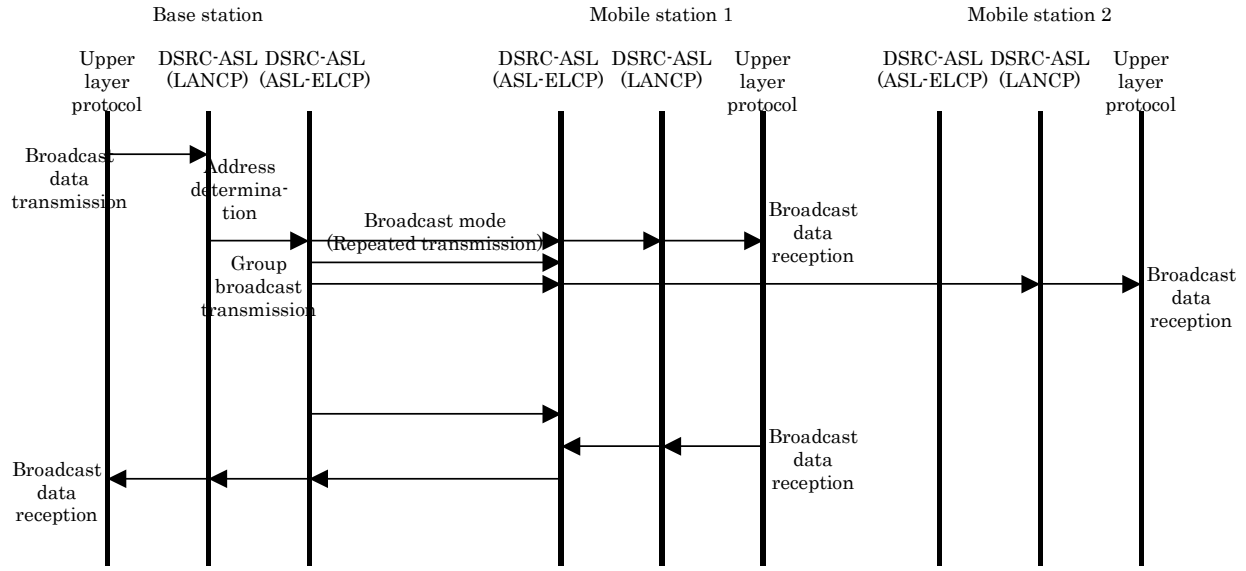
Following procedure applies to the case that the communication frame destination MAC address, which passes through during the communication phase, is a unicast address.

(1) Data transmission process
When it receives the NCP-SDU from upper layer protocols, it transfers based on data transfer message.

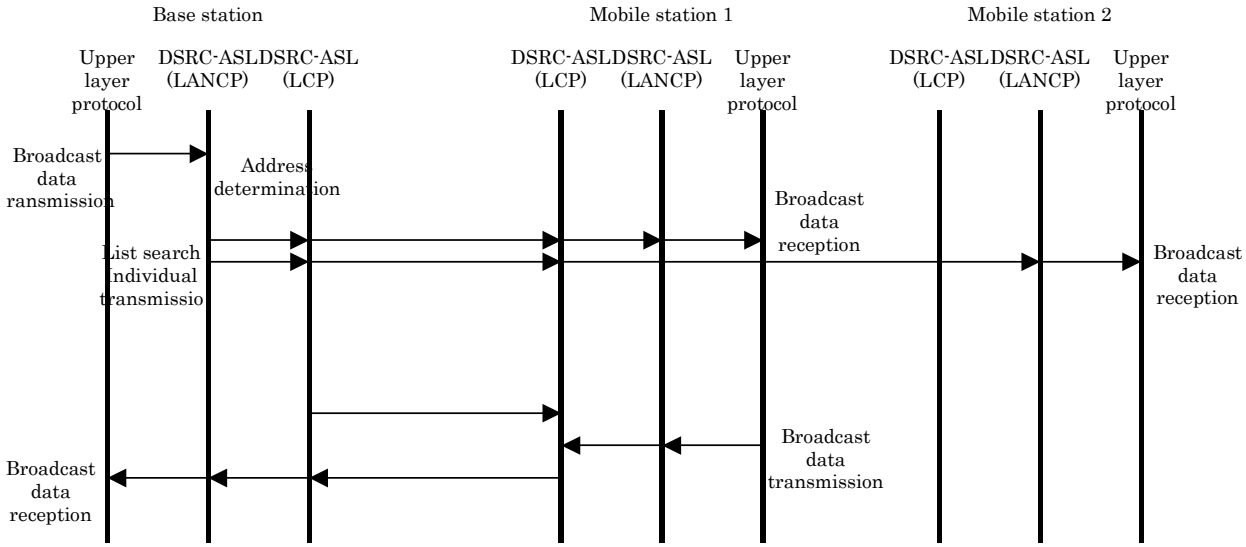
In this case, it refers the destination MAC address from the NCP-SDU for the destination link address, and obtains from the management table.

(2) Data reception process
When it receives a data transfer message, it extracts the NCP-SDU from the message and transfers it to upper layer protocols.

If it receives a message other than the data transfer message during the communication phase, the message shall be discarded.



(a) Data distribution through broadcast mode



(b) Data distribution through individual transmission

Figure 3.3-14 — Broadcast Transfer Procedure Outline

3.3.4.4.2.1.2 Broadcast Communication Process

Broadcast communication process is applied to the case that the communication frame destination MAC address, which passes through during the communication phase, is a broadcast address.

(1) Data transmission process
When the NCP-SDU, whose broadcast address is specified in the MAC address, is passed from upper layer protocols, it is transferred based on data transfer message.
In this case, the transmission method of the data transfer message shall be selected from the following two procedures.

(a) The destination link address shall be made into multicast address and transmitted as shown in Figure 3.3-14 (a).

(b) The data is sent to all the link addresses registered in the management table as shown in Figure 3.3-14 (b).

3.3.4.4.2.2 Mobile Station Data Transfer Process

(1) Data transmission process
When the NCP-SDU is passed from upper layer protocols, it is transferred based on data transfer message.

In this case, it refers the destination MAC address from the NCP-SDU for the destination link address, and obtains from the management table.

(2) Data reception process
When it receives a data transfer message, it extracts the NCP-SDU from the message and transfers it to upper layer protocols.

If it receives a message other than the data transfer message during the communication phase, the message shall be discarded.

3.3.5 PPP Control Protocols (PPPCP)

3.3.5.1 Overview

The PPPCP is a control protocol to establish a point-to-point protocol (PPP) connection via the DSRC. This protocol provides an interface to a PPP link, which is an octet-oriented synchronous link used to pass thorough PPP frames that a PPP exchanges.

3.3.5.2 Interface to PPPCP

3.3.5.2.1 Providing PPP Startup/Termination Events

The PPPCP provides a PPP link with the following events. The method to provide events is not specified as an implementation requirement.

Figure 3.3-15 shows a logical relationship of events between the PPP link and the PPPCP.

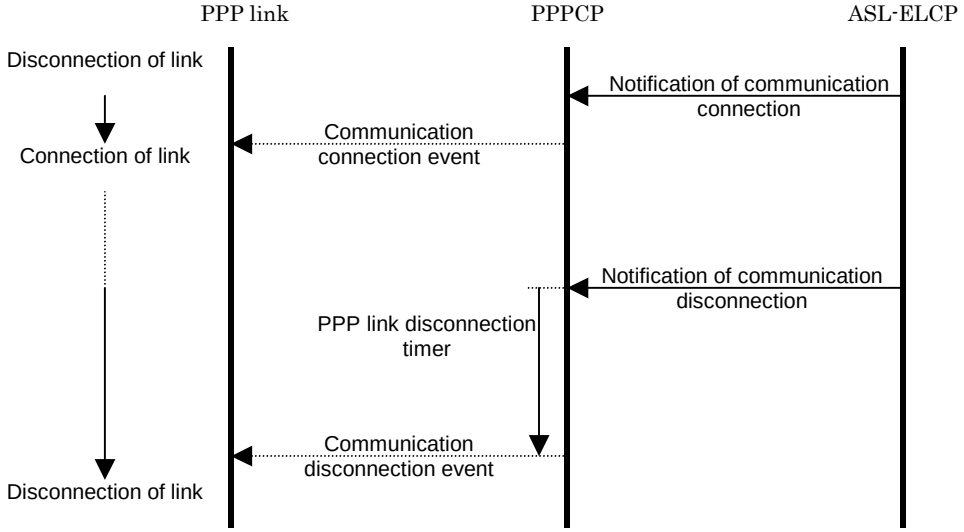


Figure 3.3-15 — Logical Relationship of PPPCP Interface

3.3.5.2.1.1 Communication Connection Event

The objective of this event is to cause a PPP link to change from the disconnected state to the connected state. This event shall be generated when a status indicating “notification of communication connection” is received from ASL-ELCP by means of event notification primitive “EventInformation.indication” of the control service. For more information about the operation while PPP link disconnection timer is in operation, see subclause 3.3.5.4.3.

3.3.5.2.1.2 Communication Disconnection Event

The objective of this event is to cause a PPP link to change to the disconnected state. This event shall be generated after a status indicating “notification of communication disconnection” has been received from the ASL-ELCP by means of event notification primitive “EventInformation.indication” of the control service, and the PPP link disconnection timer has finished operation. For more information about this operation, see subclause 3.3.5.4.3.

3.3.5.2.1.3 Communication Connection Control by External Events

This is not specified as an implementation requirement.

3.3.5.2.2 Communication Frame Transfer

A PPPCP exists between an ASL-ELCP and a PPP link, and provides a function to pass through communication frames that a PPP exchanges. (See Figure 3.3-16.)

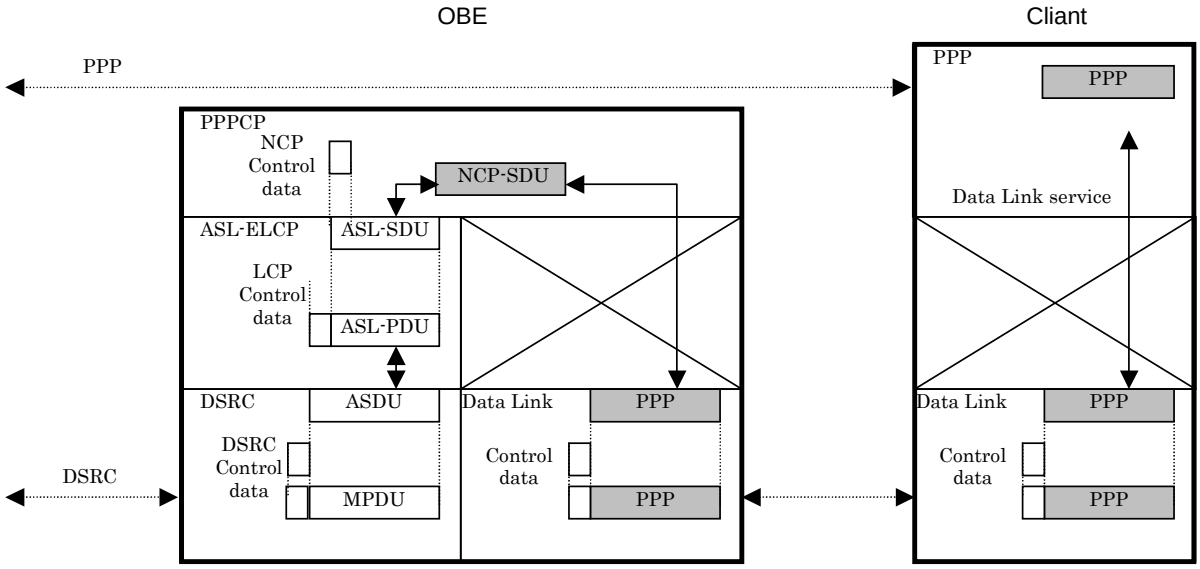


Figure 3.3-16 — Example of PPP Frame Transfer

3.3.5.3 PPPCP Procedure Element

3.3.5.3.1 Upper Layer Protocol Specification

The communication frames “PPP frames” of the upper layer protocol, which is to be passed through by a PPPCP, shall be octet-stuffed framing, which is specified by IETF RFC1662. (See Figure 3.3-17.)

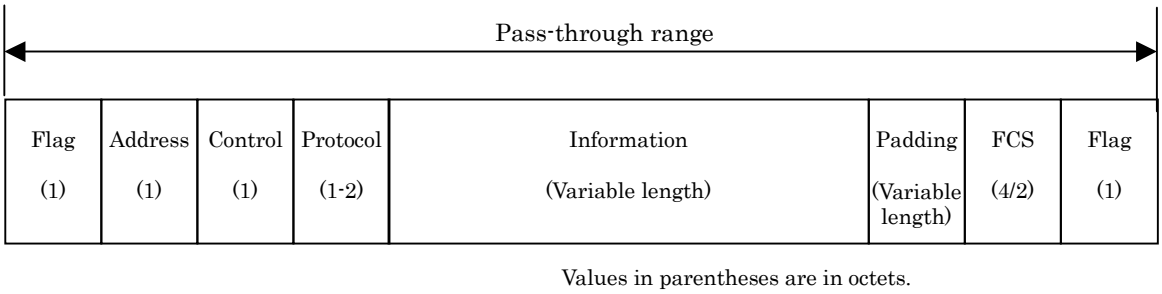


Figure 3.3-17 — Pass-through Range of Communication Frame

3.3.5.3.2 Maximum Transfer Unit (MTU) of PPPCP

The MTU means the maximum data unit length (MTU) that a PPPCP can pass to an ASL-ELCP. The MTU of the PPPCP shall be “522” octets (including access control information).

3.3.5.3.3 Access Control Information Format

The PPPCP defines massages which store control information shown in Table 3.3-14 and are shared between the mobile station and the base station in order to give instructions on the PPPCP procedure in access control information.

Table 3.3-14 — PPPCP Protocol Identifiers

Protocol identifier	Description	Option field format (See Annex B.)
0	Data transfer	None
1	Status notification	PppStatus type
2	Command notification	PppCommand type
3-15	Reserved	

3.3.5.3.3.1 Data Transfer Message

This is a massage used to transfer the upper layer protocol data unit (PDU). The data transfer message format is as shown in Table 3.3-15.

Table 3.3-15 — Data Transfer Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier pppContol (3)				Protocol identifier message(0)			

(1) Access point identifier
An identifier indicating PPPCP “pppContol (3)” is stored.

(2) Protocol identifier
An identifier indicating data transfer “message (0)” is stored.

(3) Option field of protocol identifier
Not used.

3.3.5.3.3.2 Status Notify Message

This is a message that the mobile station sends to the base station to notify the base station of the PPP link status of the mobile station. The status notification message format is as shown in Table 3.3-16.

Table 3.3-16 — Status Notify Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier pppContol (3)				Protocol identifier status (1)			
2	Status value							

(1) Access point identifier
An identifier indicating PPPCP “pppContol (3)” is stored.

(2) Protocol identifier
An identifier indicating status notification “status (1)” is stored.

(3) Option field of protocol identifier
As an encoding result of “PppStatus” type, the following contents are stored.

(a) Status value
This is an identifier that indicates the PPP link status of the mobile station. For more details, see Table 3.3-17.

Table 3.3-17 — Status Value Details

Value	Identifier	Description
0	newLink	Initial Association status of PPP link
1	continuousLink	Reconnected status of PPP link
2	disconnected	Disconnected status of PPP link
3	commandError	Command error
4-255		Reserved

3.3.5.3.3.3 Command Notify Message

This is a message that the base station sends to the mobile station to synchronize the PPP link status between the base station and the mobile station. The command notification message format is as shown in Table 3.3-18.

Table 3.3-18 — Command Notify Message Format

	7 (MSB)	6	5	4	3	2	1	0 (LSB)
1	Access point identifier pppContol (3)				Protocol identifier command (2)			
2	Command code value (commandCode)							
3	Command parameter (commandParameter)							

(1) Access point identifier
An identifier indicating PPPCP “pppContol (3)” is stored.

(2) Protocol identifier
An identifier indicating command notification “command (2)” is stored.

(3) Option field of protocol identifier
As an encoding result of PppCommand format, the following contents are stored:

(a) Command code value (commandCode)
This is an identifier that controls the PPP link status of the mobile station. For more details, see Table 3.3-19.

(b) Command parameter (commandParameter)
This is additional information added to a command code. The contents of additional information are defined by the variable of CommandParameter format “parameter”. (See Annex B.)

The relationship between the identifier of a command code and the identifier of a variable “parameter” is as shown in Table 3.3-19.

Table 3.3-19 — Command Code Value and Command Parameter

Command code value			Command parameter
Value	Identifier	Description	Identifier of variable “parameter”
0	connectPpp	Request for connection of PPP link	pppLinkTime
1	disconnectPpp	Request for disconnection of PPP link	null
2	continuePpp	Request for continued connection of PPP link	null
3-255		Reserved	

3.3.5.3.4 Base Station Control Table

The base station control table is a table for matching link address and identification information, which is set at the interface to a PPP link, in order to sent and receive the NCP-SDU to and from the PPP link.

This control table is generated when a status “notification of communication connection” is received by means of event notification primitive “EventInformation.indication” of the control service, and discarded when a status “notification of communication disconnection” is received.

3.3.5.3.5 PPP Link Disconnection Timer

The PPP link disconnection timer is a timer to maintain a PPP link connection for a set period of time. This timer is generated when a status “notification of communication connection” is received by means of event notification primitive “EventInformation.indication” of the control service, and discarded when a status “notification of communication disconnection” is received.

The link disconnection timer value is determined by the base station and designated to the mobile station by the command parameter “pppLinkTime” in a command notification message, of which command code value is “connectPpp”.
The timer value shall be determined according to base station installation conditions, etc.
NOTE This timer is intended to reduce the frequency of PPP link connection and disconnection operations in a situation where DSRC connection and disconnection operations can be conducted frequently.

3.3.5.4 PPPCP Procedure

3.3.5.4.1 Connection Process Procedure

3.3.5.4.1.1 Initialization

(1) Initialization of base station
If base station PPPCP has received a status “notification of communication connection” by means of the event notification primitive “EventInformation.indication” of the control service, it generates a control table for destination control based on the link address passed by this primitive and registers the identification information set at the interface to the PPP link.

The specifications for identification information to be set at the interface to the PPP link and the method to obtain such information are not specified as implementation requirements.

After generating a control table, the base station PPPCP waits a status notification message

from the mobile station. If it has received a status notification message from the mobile station, it determines the start of communication phase as specified in subclause 3.3.5.4.1.3.

(2) Initialization of mobile station
If the mobile station PPPCP has received a status “notification of communication connection” by means of the event notification primitive “EventInformation.indication” of the control service, it stores a value corresponding to the PPP link status of the mobile station in a variable “pppStatus” and sends a status notification message.

After having sent the message, mobile station PPPCP waits a command notification message from the base station. If it has received a command notification message from the base station, it determines the start of communication phase as specified in subclause 3.3.5.4.1.3.

3.3.5.4.1.2 Communication Termination Process

(1) Communication termination processing of the base station
If base station PPPCP has received a status “notification of communication disconnection” by means of the event notification primitive “EventInformation.indication” of the control service, it determines the termination of communication phase according to the procedure specified in subclause 3.3.5.4.3. If the termination of communication phase is selected, the control table of the link address passed by the above mentioned primitive is discarded.

(2) Communication termination processing of mobile station
If mobile station PPPCP has received a status “communication disconnection” by means of event notification primitive “EventInformation.indication” of the control service, it determines the termination of communication phase according to the procedure specified in subclause 3.3.5.4.3.

3.3.5.4.1.3 PPPCP Connection Process

If a status “notification of communication connection” is received by means of the event notification primitive “EventInformation.indication” of the control service, the base station PPP link can be in either of the following two statuses: the PPP link disconnected status, where the link to this particular mobile station is disconnected, or the PPP link connected status, where the link to this particular mobile station is maintained. The operation of the base station PPPCP and the operation of the mobile station PPPCP in response to this base station PPPCP operation are defined by the combination of this base station PPP link status and the status indicated by the status value of the status notification message that the mobile station sent to the base station, as described below.

(1) When the base station’s PPP link is in a disconnected status and the status value is “newLink”.
If the base station has received a status notification message with its status value “newLink”, it sends a command notification message with its command code value “connectPpp” to request connection of the mobile station PPP link. At the same time, it changes to a

communication phase. It notifies the base station PPP link of “communication connection event” on this occasion.

Upon receiving a command notification message with its command code value “connectPpp”, the mobile station changes to a communication phase. It notifies the mobile station’s PPP link of “communication connection event” on this occasion.

(2) When the base station PPP link is in a disconnected status and the status value is “continuousLink”
If the base station has received a status notification message with its status value “continuousLink”, it sends a command notification message with its command code value “disconnectPpp” to request disconnection of the mobile station’s PPP link. It does not notify the base station’s PPP link of “communication connection event” on this occasion.

Upon receiving a status notification message with its command code value “disconnectPpp”, the mobile station notifies its PPP link of “communication disconnection event”. (See Figure 3.3-18.)

(3) When the base station PPP link is in a connected status and the status value is “newLink”.
If the base station has received a status notification message with its status value “newLink”, it sends a command notification message with its command code value “disconnectPpp”. It notifies the base station PPP link of “communication disconnection event” on this occasion.

Upon receiving a status notification message with its command code value “disconnectPpp”, the mobile station notifies its PPP link of “communication disconnection event”. (See Figure 3.3-19.)

(4) When the base station PPP link is in a connected status and the status value is “continuousLink”.
If the base station has received a status notification message with its status value “continuousLink”, it sends a command notification message with its command code value “connectPpp” to request continued connection of the mobile station’s PPP link. At the same time, it changes to a communication phase. It does not notify the base station’s PPP link of “communication connection event” on this occasion.

Upon receiving a status notification message with its command code value “continuePpp”, the mobile station changes to a communication phase. It does not notify the mobile station’s PPP link of “communication connection event” on this occasion.

(5) Cases except for (1) – (4)
If the base station has received a status notification message with a status value other than those mentioned above, it sends a command notification message with its command code value “disconnectPpp”. It notifies the base station PPP link of “communication disconnection event” on this occasion.

Upon receiving a status notification message with its command code value “disconnectPpp”, the mobile station’s PPPCP notifies its PPP link of “communication disconnection event”.

NOTE When the base station sends a command notification message with its command code value “disconnectPpp” to request the mobile station to disconnect the PPP link, it is preferable that the base station disconnects DSRC and starts the mobile station release timer. The purpose is to avoid DSRC congestions until the mobile station starts reconnecting a PPP link. The mobile station is supposed to start reconnecting DSRC after the release timer exceeds a set period of time. It should be noted that there could be a case where other ASL-NCP and DSRC applications are being started at the same time.

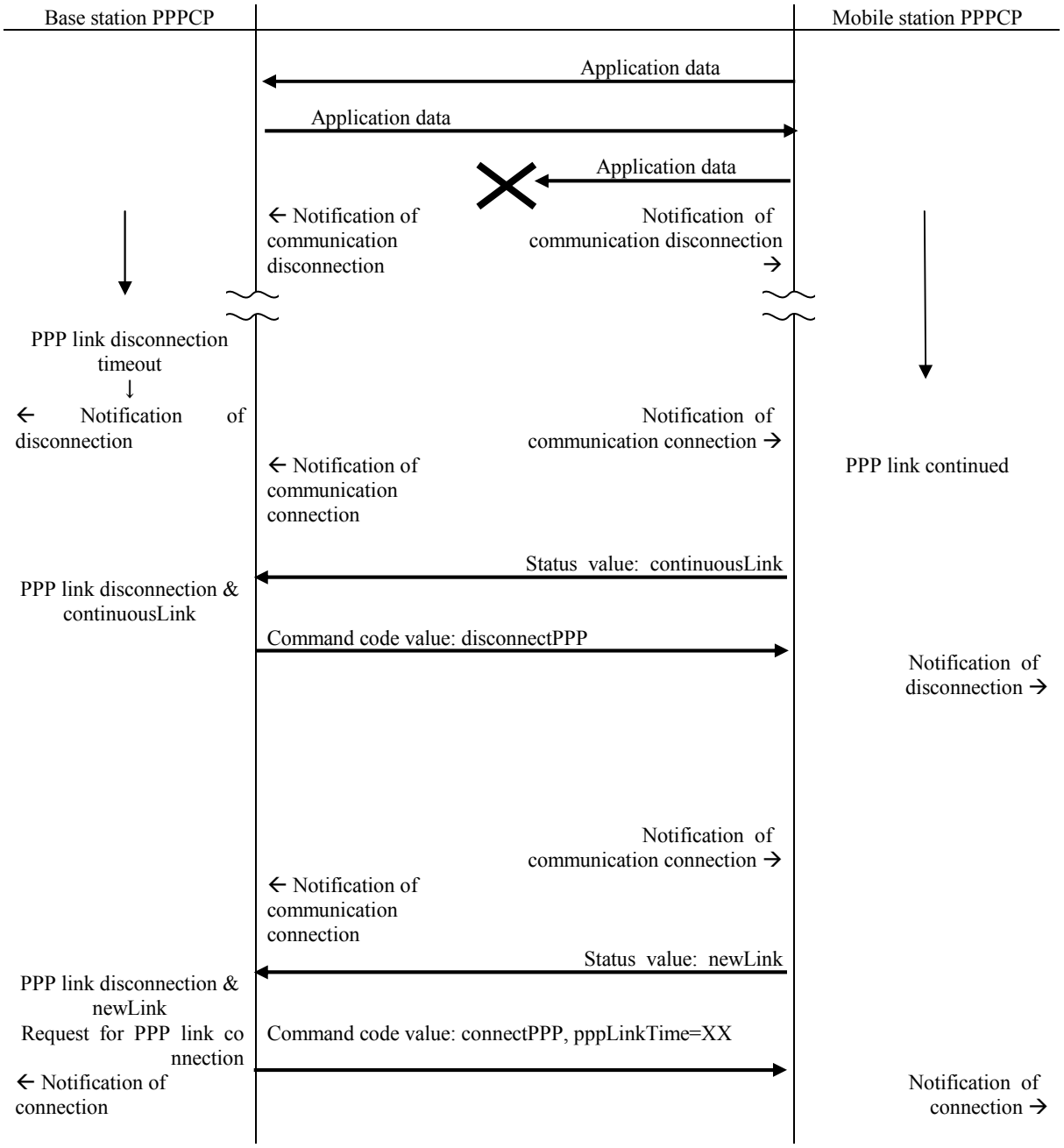


Figure 3.3-18 — Example of Reconnection Sequence (1)

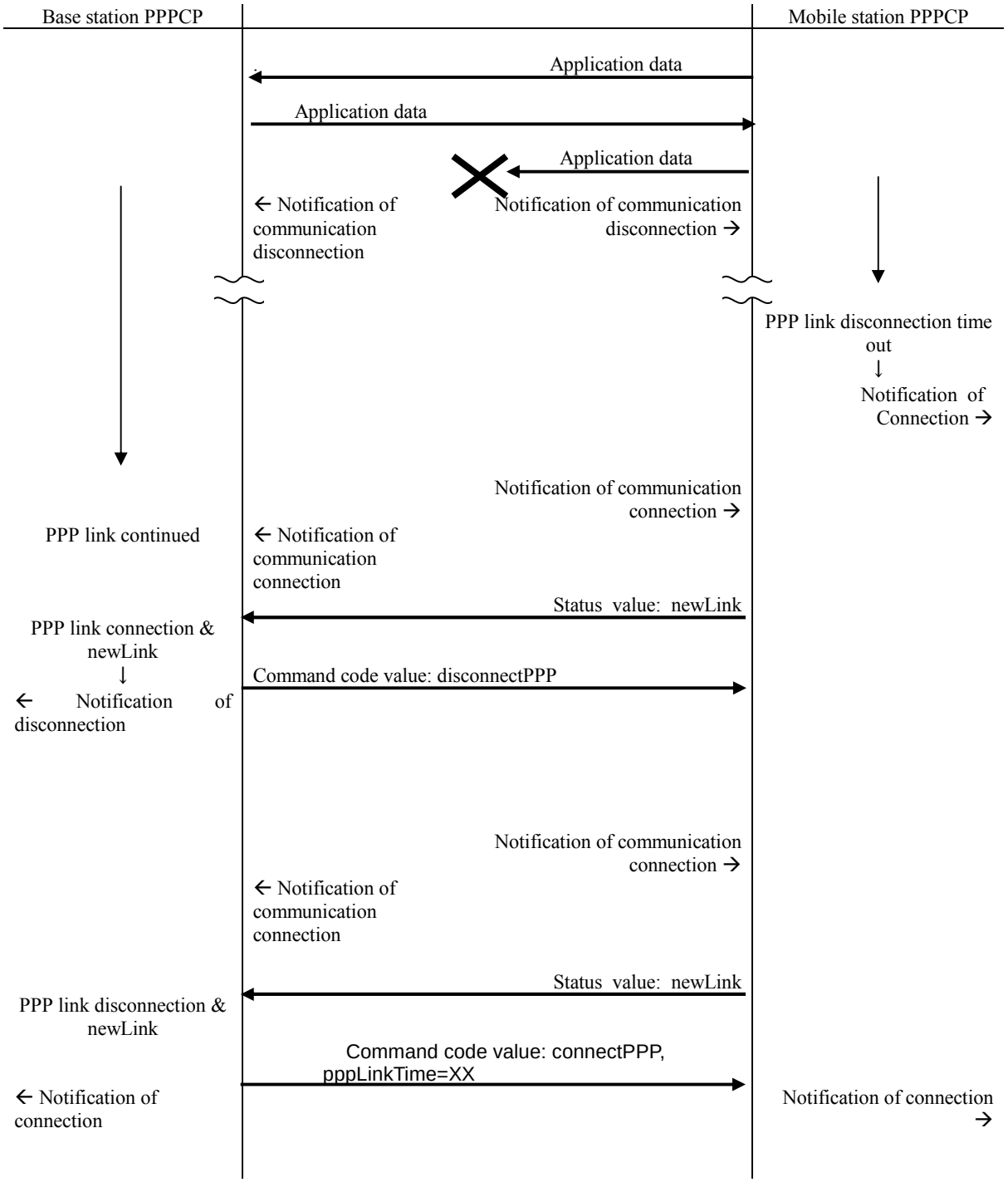


Figure 3.3-19 — Example of Reconnection Sequence (2)

3.3.5.4.2 Data Transfer Procedure

3.3.5.4.2.1 Base Station PPPCP Data Transfer Process

(1) Data transfer process

If a base station PPPCP is passed an NCP-SDU from a PPP link, it transfers the NCP-SDU by data transfer message. In such a case, it obtains a destination link address from the NCP-SDU in reference to the control table based on the identification information set at the interface to the PPP link.

(2) Data reception process
If a base station PPPCP has received the data transfer message, it extracts an NCP-SDU from the message and transfers it to the PPP link. In such a case, it refers to the control table using a link address and identifies a destination based on the identification information set at the interface to the PPP link.

If a message received during communication phase is not a data transfer message, the message is discarded.

3.3.5.4.2.2 Mobile Station PPPCP Data Transfer Process

(1) Data transfer processing
If a mobile station PPPCP is passed an NCP-SDU from a PPP link, it transfers the NCP-SDU by the data transfer message.

(2) Data reception processing
If a mobile station PPPCP has received the data transfer message, it extracts an NCP-SDU from the message and transfers it to the PPP link.

If a message received during communication phase is not a data transfer message, the message is discarded.

3.3.5.4.3 PPP Link Disconnection Timer Procedure

3.3.5.4.3.1 Setting a Link Disconnection Timer Value Setting

A base station PPPCP notifies a mobile station PPPCP of a setting time for the PPP link disconnection timer (pppLinkTime) by PPPCP connection processing. (See subclause 3.3.5.4.1.3.) The “pppLinkTime” shall be set in reference to “PppLinkTime” within “connectPPP” command. This time shall be set in 0.2-second increments. After receiving the “pppLinkTime” value, the mobile station PPPCP shall use it as a default value of the PPP link disconnection timer.

3.3.5.4.3.2 Procedure

The link disconnection timer procedure to be followed by base station PPPCP and mobile station PPPCP are as described below.

(1) If a PPPCP has received “notification of communication disconnection” by means of

event notification primitive “EventInformation.indication” of the control service, it starts the PPP link disconnection timer. The connection of the PPP link is maintained without being disconnected while this timer is in operation.

(2) If a PPPCP has received “notification of continued communication” by means of event notification primitive “EventInformation.indication” of the control service, it stops the PPP link disconnection timer. Then, subsequent processing is chosen according to the procedure specified in subclause 3.3.5.4.1.3.

(3) If a PPP link disconnection timer timeout has occurred, a PPPCP disconnects the PPP link by notifying the PPP link of “communication disconnection event”.

NOTE It is possible that a part of PPP packets are lost due to communication error, etc. while the PPP link disconnection timer is in operation. In such a case, packet recovery processing is left to the upper layer protocols such as TCP layer.

4 Abbreviation

[A]

ACCM	: Async-Control-Character-Map
AID	: Application Identifier
ASK	: Amplitude Shift Keying
ASL-ELCP	: DSRC-ASL Extended Link Control Protocol
ASL-NCP	: DSRC-ASL Network Control Protocol
ASL-PDU	: ASL Protocol Data Unit
ASL-SDU	: ASL Service Data Unit
ASN.1	: Abstract Syntax Notation One

[B]

BST	: Beacon Service Table

[C]

CHAP	: Challenge-Handshake Authentication Protocol
CTO	: Connection Timer for OBU
CTR	: Connection Timer for RSU

[D]

DHCP	: Dynamic Host Configuration Protocol
DSRC	: Dedicated Short-Range Communication
DSRC-ASL	: DSRC Application Sub-Layer

[E]

EFC	: Electronic Fee Collection
EID	: Element Identifier
ETC	: Electronic Toll Collection System

[F]

FCMC	: Frame Control Message Channel

[I]

I-KE	: Initialization Kernel Element
IID	: Invoker Identifier
IP	: Internet Protocol
ITS	: Intelligent Transport Systems

[L]

LAN	: Local Area Network
LANCP	: LAN Control Protocol
LID	: Link Identifier
LPCP	: Local Port Control Protocol
LPP	: Local Port Protocol

[M]

MAC	: Media Access Control
MDC	: Message Data Channel
MIB	: Management Information Base
MPI	: Multi Purpose Information system
MPP	: Multi Purpose Payment system
MRU	: Maximum Receiving Unit
MTU	: Maximum Transmission Unit

[N]

NCP-PDU	: ASL-NCP Protocol Data Unit
NCP-SDU	: ASL-NCP Service Data Unit

[O]

OBE	: On Board Equipment
OBU	: On Board Unit

[P]

PAP	: Password Authentication Protocol
PDU	: Protocol Data Unit
PER	: Packed Encoding Rule
PPP	: Point-to-Point Protocol
PPPCP	: PPP Control Protocol

[Q]

QPSK	: Quadrature Phase Shift Keying

[R]

RSU	: Road Side Unit

[S]

SDU	: Service Data Unit
SUU	: Segment Unit for Unicast
SUM	: Segment Unit for Multicast

[T]

T-KE	: Transmission Kernel Element
TCP	: Transmission Control Protocol

[U]

UDP	: User Datagram Protocol

[V]

VST	: Vehicle Service Table

[W]

WCNC	: Wireless Call Number Channel

WTTS : Watchdog Timer for Transmission Schedule

Empty page

5 Variables

5.1 ASL-ELCP Variables

5.1.1 ASL-ELCP communication control

k : A repetition transmitting number of the broadcast mode control
MRU : An ASL-ELCP maximum received data length passed from ASL-NCP
SUM : An ASL-SDU segment unit in the bulk transmission mode (broadcast mode)
SUU : An ASL-SDU segment unit for the unicast in the bulk transmission mode (except broadcast mode)

5.1.2 ASL-ELCP communication management

CTO,T1max : A communication connection management timer. The mobile station timer for watching the link connection condition between OBU (s) and a RSU.
CTR,T2max : A communication connection management timer. The base station timer for watching the link connection condition between OBU (s) and a RSU.
NA : A number of authentication trials
WTTS : A base station Watchdog Timer for Transmission Schedule

5.2 ASL-NCP Variables

5.2.1 LPCP Variables

MTU : An allowable maximum data unit length passing to the ASL-ELCP from the LPCP.

5.2.2 LANCP Variables

MTU : An allowable maximum transfer data unit length passing to the ASL-ELCP from the LANCP.

5.2.3 PPPCP Variables

MTU : An allowable maximum transfer data unit length passing to the ASL-ELCP from the PPPCP.
ppp link disconnection timer: A PPP link connection maintaining timer

Empty page

ANNEX

Annex A
(normative)
Protocol Parameter

A1 Extended Link Control Protocol (ASL-ELCP)

A1.1 Value of Variables using ASL-ELCP

(1) Maximum Receive Unit (MRU)
An MRU is a maximum value of MTU installed in ASL-NCP.

(2) Segment Unit for Unicast (SUU)
A SUU is defined as “56” octet length in case of the ASK system, “183” octet length in case of the π/4 shifted QPSK system.

(3) Segment Unit for Multicast (SUM)
A SUM is defined as “54” octet length in case of the ASK system, “181” length octet in case of the π/4 shifted QPSK system.

(4) A cyclic (repetition) transmission number of base station in the broadcast control mode (k)
A k is specified in consideration with the individual system reliability.

(5) A number of the authentication trials of base station (NA)
NA is not defined. This number is the system implementation dependent.

(6) A value of the Communication connection Timer for base station (CTR) (T2max)
T2max is defined by the installation condition of a base station etc.

(7) A value of the Watchdog Timer for Transmission Schedule (WTTS) of the base station (T1max)
T1max is defined by installation condition of a base station etc.

(8) A value of the Communication connection Timer for a mobile station (CTO) (T1max)
T1max is defined by installation condition of a base station etc.

A1.2 Elements of Road side unit (base station) management parameters of ASL-ELCP (informative)

```
-- base station
RoadSideManagementParameter DEFINITIONS::=
BEGIN

  IMPORTS
  RoadSideProfile, VehicleProfile     FROM   AslProfile;        -- see .Annex B
  DsrcLID                             FROM   ElcpParameter;     -- see .Annex B
```

```
-- Initial profile data of the base station
RoadSideProfileTable::= RoadSideProfile,                    -- base station profile
RoadSideManagementTable ::= SEQUENCE {
        timer-T2max             INTEGER(0..4095), -- set up value of CTR
        timer-T1max             INTEGER(0..4095), -- set up value of WTTS
        maximumReciveUnit       INTEGER(0..2047), -- value of MRU
        unicastSegmentUnit      INTEGER(0..255),  -- value of SUU
        broadcastSegmentUnit    INTEGER(0..255),  -- value of SUM
        broadcastRepeatTime     INTEGER(0..7),    -- Number of times of
                                                  -- continuation transmission
                                                  -- for Multicast(k)
        authenticationTime  INTEGER(0..7)         -- Number of authentication
                                                  -- trials (NA)
        }

-- Management parameter table of the mobile station
UserProfileTable::= SEQUENCE OF SEQUENCE{
        dsrcLinkAddress     DsrcLID,
        vehicleProfile      VehicleProfile
        }
END

-- the mobile station
VehicleManagementParameter DEFINITIONS::=
BEGIN

   IMPORTS
   RoadSideProfile, VehicleProfile      FROM    AslProfile;    -- see .Annex B

-- Initial profile data of mobile station
VehicleProfileTable::= VehicleProfile
VehicleManagementTable ::=   SEQUENCE {
        maximumReceiveUnit      INTEGER(0..2047), -- value of MRU
        unicastSegmentUnit      INTEGER(0..255),  -- value of SUU
        broadcastSegmentUnit    INTEGER(0..255),  -- value of SUM
        }

-- profileparameter of base station
StationProfileTable ::= RoadSideProfile
END
```

A2 DSRC-ASL Network Control Protocol (ASL-NCP)

A2.1 Local Port Control Protocol (LPCP)

A2.1.1 Value of Variables using LPCP

(1) Maximum Transmission Unit (MTU)
MTU is defined as “522” octets.

A2.1.2 Element of Mmanagement Parameters of LPCP (informative)

```
-- base station
RoadSideLpcpManagementParameter DEFINITIONS::=
BEGIN

  IMPORTS
  AslEquipmentID, DsrcLID     FROM   ElcpParameter;              -- see .Annex B
  PortNo                      FROM   LocalControlParameter;      -- see .Annex B


-- Initial profile data of base station
RoadsideLpcpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit      INTEGER(0..2047),      -- value of MTU
       availablePort                AvailablePortList
        }
AvailablePortList ::= SEQUENCE OF PortNo -- List of receive available Local Port


-- User Address Management Table
UserAddressManagementTable ::= SEQUENCE OF SEQUENCE{
       dsrcLinkAddress     DsrcLID,
       userEquipmentID     AslEquipmentID
       }
END


-- the mobile station
VehicleLpcpManagementParameter DEFINITIONS::=
BEGIN

  IMPORTS
  PortNo        FROM   LocalControlParameter;            -- see Annex B

-- Initial profile data of mobile station
VehicleLpcpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit INTEGER(0..2047), -- value of MTU
       availablePort           AvailablePortList
       }
AvailablePortList ::= SEQUENCE OF PortNo      -- List of receive available Local Port
END
```

A2.2 LAN Control Protocol (LANCP)

A2.2.1 Value of Variables using LANCP

(1) Maximum Transmission Unit (MTU)
MTU is defined as “1517” octets.

A2.2.2 Element of Management Parameters of LANCP (informative)

```
-- base station
RoadSideLancpManagementParameter DEFINITIONS::=
BEGIN

  IMPORTS
  AslEquipmentID, DsrcLID      FROM   ElcpParameter;          -- see Annex B
  InitialData                  FROM   LanControlParameter;        -- see Annex B

-- Initial profile data of base station
RoadsideLancpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit INTEGER(0..2047),      -- value of MTU
       serverMacAddress    InitialData                -- Server MAC Address
       }

-- User Address Management Table
UserAddressManagementTable ::= SEQUENCE OF SEQUENCE{
       dsrcLinkAddress     DsrcLID,
       userMacAddress      InitialData,               -- Client MAC Address
       userEquipmentID     AslEquipmentID
       }
END


-- mobile station
VehicleLancpManagementParameter DEFINITIONS::=
BEGIN

  IMPORTS
  InitialData                FROM   LanControlParameter;        -- see .Annex B

-- Initial profile data of mobile station
VehicleLancpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit INTEGER(0..2047),      -- Value of MTU
       clientMacAddress        InitialData            -- Client MAC Address
       }
END
```

A2.3 PPP Control Protocol (PPPCP)

A2.3.1 Value of Variables using PPPCP

(1) Maximum Transmission Unit MTU
MTU is defined as “522” octets.

(2) Link Disconnection Timer
A value of the Link Disconnection Timer is defined by the base station implementation condition, etc.

A2.3.2 Elements of Management Data of PPPCP (informative)

```
-- the base station
RoadSidePppcpManagementParameter DEFINITIONS::=
BEGIN
```

```
   IMPORTS
   AslEquipmentID, DsrcLID      FROM   ElcpParameter;          -- see .Annex B
   PppLinkTime                  FROM   PppControlParameter;  -- see .Annex B


-- Initial profile data of base station
RoadsidePppcpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit  INTEGER(0..2047),      -- Value of MTU
       linkHoldTime             PppLinkTime            -- Link Disconnection Timer of
                                                       -- PPP
       serverAccessIdentifier   OCTET STRING           -- Identifier of PPP
       }


-- Address Management Table
UserAddressManagementTable ::= SEQUENCE OF SEQUENCE{
       dsrcLinkAddress          DsrcLID,
       userAccessIdentifier     OCTET STRING,          -- Identifier of PPP client
       userEquipmentID          AslEquipmentID
       }
END

-- mobile station
VehiclePppcpManagementParameter DEFINITIONS::=
BEGIN

   IMPORTS
   PppLinkTime              FROM   PppControlParameter;       -- see .Annex B


-- Initial profile data of mobile station
VehiclePppcpManagementTable ::= SEQUENCE {
       MaximumTransmissionUnit  INTEGER(0..2047),      -- Value of MTU
       linkHoldTime             PppLinkTime            -- PPP Link Disconnection
                                                       -- Timer
       }
END
```

Annex B
(normative)
Data Structures

B1 Extended Link Control Protocol (ASL-ELCP)

B1.1 ASL Profile

```
AslProfile DEFINITIONS ::=
BEGIN

   IMPORTS
   ServiceTime, AccessControl, LinkControl, AslEquipmentID   FROM      ElcpParameter;

-- The base station DSRC-ASL Profile
RoadSideProfile ::= SEQUENCE {
      versionIndex    INTEGER(0..15),       -- Version of base station DSRC-ASL
      serviceTime     ServiceTime,          -- unit of lifetime value(MS), 0 is
                                            -- infinite
      accessControl   AccessControl,        -- type of supported ASL-NCP
      linkControl     LinkControl           -- function supported by ASL-ELCP
      }


-- The mobile station DSRC-ASL Profile
VehicleProfile ::= SEQUENCE {
      fill            BIT STRING(SIZE(4)),  -- value of encoding is set up 0
      versionIndex    INTEGER(0..15),       -- Version of mobile station
                                         -- DSRC-ASL
      equipmentID     AslEquipmentID,       -- Identifier of mobile station
      accessControl   AccessControl,        -- type of supported ASL-NCP
      linkControl     LinkControl           -- function supported by ASL-ELCP
      }
END


ElcpParameter DEFINITIONS ::=
BEGIN

   EXPORTS
   AccessControl, AccessControlIndex, AslEquipmentID, LinkControl, ServiceTime, DsrcLID;

-- Definition of Variables
AccessControl ::= SEQUENCE(0..255) OF AccessControlIndex
AccessControlIndex ::= INTEGER {
      aslControlManagement    (0),          -- Communication Control
                                            -- Management
      localPortControl        (1),          -- Local Port
      lanControl              (2),          -- IEEE 802.3 MAC
      pppControl              (3),          -- PPP
      localPortControl2       (14),         -- Local Port (for Road Enterprise Use)
      octetAlignment          (15)          -- for Octet Alignment Use
   -- AccessControlIndex Control index 4 to 13 is reserved
       } (0..15)
   AslEquipmentID ::= OCTET STRING(SIZE(6))       -- 12 digit number of BCD code
                                                  -- see Annex F
```

```
DsrcLID ::= OCTET STRING(SIZE(4))
LinkControl ::= SEQUENCE(0..255) OF LinkControlFunction
LinkControlFunction ::= SEQUENCE {
      fill                BIT STRING(SIZE(4)), --value of encoding is set up 0
      lcpFunction         FunctionTags
      }
FunctionTags ::= CHOICE {
      notUse              [0]      NULL,
      secureFunction      [1]      ProfileSecurity,  -- Security
      bulkTransmit        [2]      NULL,             -- Bulk transmission
      broadCast           [3]      NULL,             -- Broadcast
      handOver            [4]      NULL,             -- Handover(reserved)
      dummy5              [5]      NULL,
      dummy6              [6]      NULL,
      dummy7              [7]      NULL,
      dummy8              [8]      NULL,
      dummy9              [9]      NULL,
      dummy10             [10]     NULL,
      dummy11             [11]     NULL,
      dummy12             [12]     NULL,
      dummy13             [13]     NULL,
      dummy14             [14]     NULL,
      dummy15             [15]     NULL
      -- Id and Definition of function tag number 4 to 15 is reserved.
      }
ProfileSecurity ::= SEQUENCE{
      fill       BIT STRING(SIZE(2)),           -- value of encoding is set up 0
      authenticate        BOOLEAN,              -- authenticate Identifier (true:
                                                -- valid/false:invalid)
      encryption          BOOLEAN,              -- encryption Identifier (true:
                                                -- valid/false:invalid)
      macOption           BOOLEAN,              -- certification Identifier
                                                -- (true:valid/false:invalid)
      userID              UserIdentifier,       -- Equipment Identifier(userID)
      authenticateMethod IA5String(SIZE(4..255)) OPTIONAL, -- Algorithm identifier
      encryptionMethod   IA5String(SIZE(4..255)) OPTIONAL, -- Algorithm identifier
      macMethod          IA5String(SIZE(4..255)) OPTIONAL   -- Algorithm identifier
      -- It is supposed that character contains distinguishable more than 4 characters
      -- to less than 255 characters when algorithm identifier is used. It should be
      -- considered data size doesn't become huge.
      -- encryption、macOption are reserved for use in the future.
      }
ServiceTime ::= INTEGER(0..4095)
UserIdentifier ::= OCTET STRING(SIZE(0..255))


-- Dependent Variables (for Interface to ASL-NCP use)
UserProfile ::= SEQUENCE {
      linkAddress    DsrcLID,
      equipmentID    AslEquipmentID        -- Identifier of mobile station
      }
END
```

B1.2 Format of Protocol Data Unit (PDU)

```
AslProtocolDataUnit DEFINITIONS ::=
BEGIN

  IMPORTS
```

```
   ServiceTime             FROM     ElcpParameter;
   NetworkControlPDU       FROM     NcpProtocolDataUnit;

-- PDU format
AslPDU ::= SEQUENCE {
       aslLinkProtocol AslLinkProtocol,       -- ASL-ELCP Control data
       aslPduBody NetworkControlPDU           -- SDU from ASL-NCP
       }


-- ASL Link Protocol data
AslLinkProtocol ::= SEQUENCE {
       BulkEnable      BOOLEAN,               -- true: bulk transmission valid/false
                                              -- : invalid)
       bulkTermination BOOLEAN,               -- true: last bulk segment
                                              -- /false bulk segment is continued
       pduGroup        INTEGER(0..31),        -- Bulk transmission Group Identifier
       segmentNumber   INTEGER(0..255),       -- Segment number
       broadcastParameter  ProfileBroadcast   OPTIONAL
       }
ProfileBroadcast ::= SEQUENCE {
       dummy      BIT STRING(SIZE(4)),        -- value of encoding is set up 0
       serviceTime     ServiceTime            -- Value of Link Connection control Timer
       }
END
```

B1.3 Link Control Management

```
LinkManagementParameter DEFINITIONS ::=
BEGIN

   EXPORTS
   LinkSubProtocol;

   IMPORTS
   AccessControlIndex      FROM     ElcpParameter;

LinkSubProtocol ::= CHOICE {
       notUse          [0]      NULL,
       echo            [1]      MsEchoParameter,       -- echo service
       echoReply       [2]      MsEchoParameter,       -- echo service
       eventReport     [3]      MsEventParameter,      -- event report service
       challenge       [4]      MsAuthCodeChallenge,   -- Access control service
                                                       -- (random number)
       signature       [5]      MsAuthCodeSignature,   --Access control
                                                       -- service (signature)
       dummy6          [6]      NULL,
       dummy7          [7]      NULL,
       dummy8          [8]      NULL,
       dummy9          [9]      NULL,
       dummy10         [10]     NULL,
       dummy11         [11]     NULL,
       dummy12         [12]     NULL,
       dummy13         [13]     NULL,
       dummy14         [14]     NULL,
       dummy15         [15]     NULL
       --Id and Definition of LinkSubProtocol function tag number 6 to 15 is reserved.
       }
```

```
-- Echo
MsEchoParameter ::= SEQUENCE {
        dummy       BIT STRING (SIZE(4)),
        source      AccessControlIndex,
        message     OCTET STRING
        }


-- EventReport
MsEventParameter ::= SEQUENCE {
        status              MsStatusCode,
        extentionParameter  OCTET STRING OPTIONAL
        }
MsStatusCode ::= INTEGER {
        Disabled to use                           (0),
        No Access point exist                     (1),     --for opposite station
        Not available this function               (2),     --for own/opposite station
        Not is available this sub-protocol        (3),     --for opposite station
        Data size exceeds upper limit             (4),     --for own station
        No space in own transmit queue
        Transmission servise is abandoned         (5),     --for own station
        Invalid designated broadcast link address (6),     --for own station
        Not corresponding designated version      (7),     --for base station
        Access permitted                          (94),    --for mobile station
        Access denied                             (95),    --for mobile station
        Report of Connection                      (96),    --for own station
        Report of Disconnection                   (97)     --for own station
        -- Value of MsStatusCode type 8 to 93 is reserved
        -- Value of MsStatusCode type 98 to 127 is reserved
        } (0..127)


-- Access Management
MsAuthCodeChallenge ::= OCTET STRING(SIZE(0..255))        -- Challenge Data
MsAuthCodeSignature ::= OCTET STRING(SIZE(0..255))        -- Signature Data

END
```

B2 ASL Network Control Protocol (ASL-NCP)

B2.1 Format of Protocol Data Unit

```
NcpProtocolDataUnit DEFINITIONS ::=
BEGIN

  EXPORTS
  NetworkControlPDU;

  IMPORTS
  LinkSubProtocol         FROM    LinkManagementParameter;
  LocalPortSubProtocol    FROM    LocalControlParameter;
  LanSubProtocol          FROM    LanControlParameter;
  PppSubProtocol          FROM    PppControlParameter;

NetworkControlPDU ::= SEQUENCE {
        aslAccessProtocol   AslAccessProtocol,  -- Access Control Information of ASL-NCP
```

```
        pduBody             OCTET STRING     -- SDU from application
        }


-- Access Control Information
AslAccessProtocol ::= CHOICE {
        linkControlManagement   [0]      LinkSubProtocol,
        localPortControl        [1]      LocalPortSubProtocol,
        lanControl              [2]      IpSubProtocol,
        pppControl              [3]      PppSubProtocol,
        dummy4                  [4]      NULL,
        dummy5                  [5]      NULL,
        dummy6                  [6]      NULL,
        dummy7                  [7]      NULL,
        dummy8                  [8]      NULL,
        dummy9                  [9]      NULL,
        dummy10                 [10]     NULL,
        dummy11                 [11]     NULL,
        dummy12                 [12]     NULL,
        dummy13                 [13]     NULL,
        localPortControl2       [14]     LocalPortSubProtocol,
        notUse                  [15]     NULL
        -- Value of AslAccessProtocol type Tag number 4 to 13 is reserved.
        -- Size of ASN.1 type defining each tag of AslControlProtocol type is 4 bit in
        -- a rule.
        -- In future extension the result of coding AslControlProtocol type tag number
        -- coincides the value of AccessControlIndex type not to confuse.
  }
END
```

B2.2 Local Port Control Protocol (LPCP)

```
LocalControlParameter DEFINITIONS ::=
BEGIN

  EXPORTS
  LocalPortSubProtocol, PortNo;

  IMPORTS
  DsrcLID FROM      ElcpParameter

LocalPortSubProtocol ::= CHOICE {
        eventReport     [0]      LpcpEventParameter,    -- Event report service
        message         [1]      LpcpTransferDataPDU,   -- Transfer service
        dummy2          [2]      NULL,
        dummy3          [3]      NULL,
        dummy4          [4]      NULL,
        dummy5          [5]      NULL,
        dummy6          [6]      NULL,
        dummy7          [7]      NULL,
        dummy8          [8]      NULL,
        dummy9          [9]      NULL,
        dummy10         [10]     NULL,
        dummy11         [11]     NULL,
        dummy12         [12]     NULL,
        dummy13         [13]     NULL,
        dummy14         [14]     NULL,
        dummy15         [15]     NULL
        -- Id and Definition of LocalPortSubProtocol type Tag number 2 to 15 is reserved.
        }
```

```
-- Transfer service
PortNo ::= INTEGER(0..65535)
LpcpTransferDataPDU ::= SEQUENCE {
        sourcePort          PortNo,          -- Source Local Port No.
        destinationPort     PortNo           -- Destination Local Port No.
        }


-- Event Report Service
LpcpEventParameter ::= SEQUENCE {
        eventCode       LpcpEventCode,       -- Event Code
        extentionParameter  OCTET STRING     -- Event extension parameter
        }
LpcpEventCode ::= INTEGER {
        Disabled to use 0                          (0),
        Disabled to use 1                          (1),
        Disabled to use 2                          (2),
        Disabled to use 3                          (3),
        Data size exceeded the upper limit         (4),
        Transmission service is cancelled          (5),
        Invalid designated broadcast link address (6),
        Disabled to use 94                         (94),
        Disabled to use 95                         (95),
        Communication connection notice            (96),
        Communication disconnection notice         (97),
        DSRC isn't connected                       (128),   -- for own station
        Destination of Local Port is invalid       (129),   -- for peer station
        Receive available port list                (130)    -- for peer station
        -- value of LpcpEventCodel 7 to 93 is reserved.
        -- value of LpcpEventCodel 98 to 127 is reserved.
        -- value of LpcpEventCodel 131 to 255 is reserved.
        } (0..255)
LpcpPrimitiveType ::= INTEGER {
        All primitive                          (0),
        Data transmission indicate primitive   (1),
        Event report primitive                 (2),
        notUse                                 (3)
        }
InvalidPort ::= SEQUENCE {
        sourcePort      PortNo,       -- Source Local Port No.
        invalidPort     PortNo,       -- Invalid Destination Local Port No.
        lid             DsrcLID       -- LID of DSRC
        }
PortList ::= SEQUENCE OF PortNo
END
```

B2.3 LAN Control Protocol (LANCP)

```
LanControlParameter DEFINITIONS ::=
BEGIN

  EXPORTS
  LanSubProtocol;

LanSubProtocol ::= CHOICE {
        message             [0]      NULL,         -- transmission data
        obuInitialMessage   [1]      InitialData,  -- mobile station initial data
```

```
        rsuInitialMessage   [2]      InitialData, -- base station initial data
        dummy3              [3]      NULL,
        dummy4              [4]      NULL,
        dummy5              [5]      NULL,
        dummy6              [6]      NULL,
        dummy7              [7]      NULL,
        dummy8              [8]      NULL,
        dummy9              [9]      NULL,
        dummy10             [10]     NULL,
        dummy11             [11]     NULL,
        dummy12             [12]     NULL,
        dummy13             [13]     NULL,
        dummy14             [14]     NULL,
        dummy15             [15]     NULL
        -- Id and Definition of LanSubProtocol type Tag number 3 to 15 is reserved.
        }
InitialData ::= OCTET STRING(SIZE(6))                -- MAC Address
END
```

B2.4 PPP control protocol (PPPCP)

```
PppControlParameter DEFINITIONS ::=
BEGIN

  EXPORTS
  PppSubProtocol;

PppSubProtocol ::= CHOICE {
        message         [0]      NULL,
        status          [1]      PppStatus,
        command         [2]      PppCommnad,
        dummy3          [3]      NULL,
        dummy4          [4]      NULL,
        dummy5          [5]      NULL,
        dummy6          [6]      NULL,
        dummy7          [7]      NULL,
        dummy8          [8]      NULL,
        dummy9          [9]      NULL,
        dummy10         [10]     NULL,
        dummy11         [11]     NULL,
        dummy12         [12]     NULL,
        dummy13         [13]     NULL,
        dummy14         [14]     NULL,
        dummy15         [15]     NULL
        -- Id and Definition of PppSubProtocol type Tag number 3 to 15 is reserved.
        }
  PppStatus ::=   INTERGER{
        newLink             (0),     --PPP initial connection state
        continuousLink      (1),     --PPP continuous state
        disconnected        (2),     --PPP disconnect state
        commandError        (3)      --command error
        --value of PppSubProtocol type 4 to 255 is reserved.
        } (0..255)
PppCommand ::=   SEQUENCE {
        commandCode             CommandCode,      -- CommandCode
        commandParameter        CommandParameter  -- CommandParameter
        }
CommandCode ::= INTERGER{
        connectPpp          (0),     --request of PPP Link Connection
        disconnectPpp       (1),     --request of PPP Link Disconnection
        continuePpp         (2)      --request of PPP Link Continuation
```

```
        -- Value of CommandCode type 3 to 255 is reserved.
        } (0..255)
CommandParameter::= SEQUENCE {
        dummy       BIT STRING(SIZE(4)),
        parameter   ParameterIndex
        }
ParameterIndex::= CHOICE {
        null            [0]     NULL,
        pppLinkTime     [1]     PppLinkTime,
        dummy2          [2]     NULL,
        dummy3          [3]     NULL,
        dummy4          [4]     NULL,
        dummy5          [5]     NULL,
        dummy6          [6]     NULL,
        dummy7          [7]     NULL,
        dummy8          [8]     NULL,
        dummy9          [9]     NULL,
        dummy10         [10]    NULL,
        dummy11         [11]    NULL,
        dummy12         [12]    NULL,
        dummy13         [13]    NULL,
        dummy14         [14]    NULL,
        dummy15         [15]    NULL
        -- Id and Definition of ParameterIndexl type Tag number 2 to 15 is reserved.
        }
PppLinkTime ::= INTEGER(0..255)     --PPP Link Disconnection Time
END
```

B3 Relationship between Module references (informative)

It shows the relationship of module reference, which is defined in this STD in Figure B3-1 for reference.

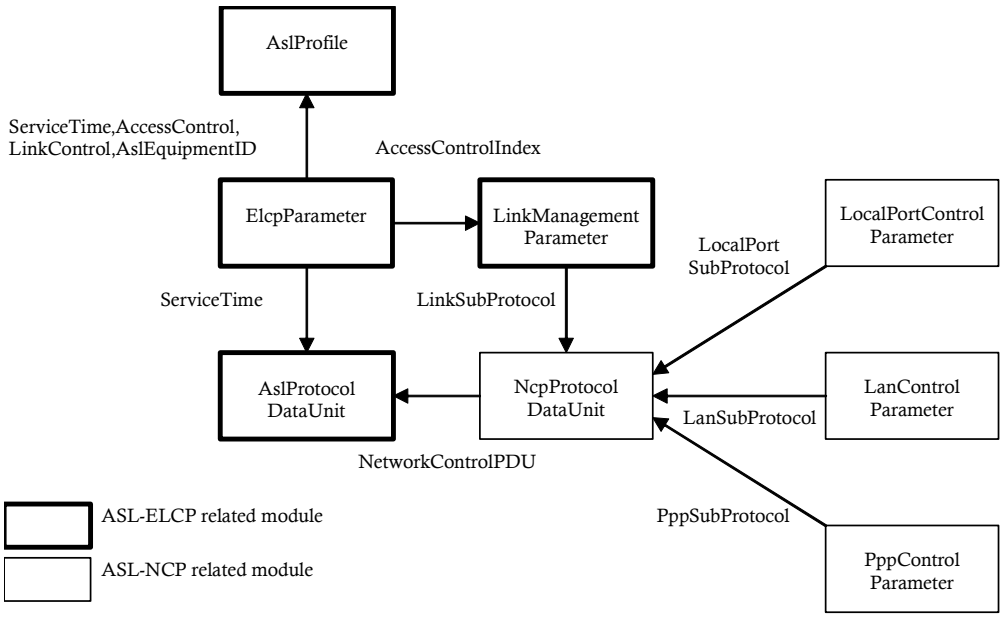


Figure B3-1 — References Relationship among Modules

Annex C
(informative)
Example of DSRC Application Sub-Layer

Table C-1 shows DSRC-ASL applicable system.

Table C-1 — DSRC-ASL Applicable System

Application		AID=18 DSRC-ASL			AID=17	AID=14	AID=1
		IP terminal	Non-IP terminal				
			separated	Tag			
Road Traffic Information	Request type navigation system	+					
	Optimum Route Guidance	+					
	Parking Lot Vacant Information and Reservation	+					
	Exchange route information between cars	+					
	Road geographic Information Automatic Renewal	+					
	Destination weather information	+					
	Calculation of Traffic jam and required time	+					
	Pedestrian route guidance	+	+				+
ETC・DSRC	ETC					+	
	Parking lot management system	+	+				+
	Car Ferry Check-in system	+	+				+
	Convenience Store Drive Through system	+					
	Logistics Management			+			+
	Gas Station Fee	+	+				+
	Vehicle Toll Gate Management			+			+
	Payment using IC card	+	+				+
Car Multi-media	Destination Information	+					
	Use for reservation	+					
	Shopping in the car	+					
	Agent in the car	+					
	Amusement Information	+					
	Internet in the car	+					
	Vehicle to vehicle communication	+					

Table C-1 — DSRC-ASL applicable system (continued)

application		AID=18 DSRC-ASL			AID=17	AID=14	AID=1
		IP terminal	Non-IP terminal				
			separated	Tag			
Logistics Public Transportation	Commercial vehicle optimum allocation system	+	+				+
	Vehicle operation management system			+			+
	Commercial delivery management system			+			+
	Total commercial delivery management system						+
	Total logistics management system	+	+				+
	Public transportation information system						+
	Public transportation vehicle management		+				+
	Co-useable type Short distance personal transportation system		+				+
Assist-cruise, Safety drive	Cross section/junction alert system				+		
	Road environment information system				+		

NOTE1 IP terminal: A radio modem and an application processor are separated, and this type OBE described in the Annex D applies to the connection with the network (e.g. IP).

NOTE2 Separated: A radio modem and an application processor are separated, and this type OBE described in the Annex D applied to non-network (non-IP) applications.

NOTE3 Tag: A radio modem and an application processor are combined in a unit, and this type OBE described in the Annex D is a simple type of the mobile station.

Annex D
(informative)
Relationship of Category of Mobile Station Function and DSRC-ASL Profile

This section shows classification of the mobile station defined by this standard and shows the applicable scope of DSRC-ASL

D1 Category of the Mobile Station

The DSRC-ASL applicable mobile station is classified into the following 4 categories in the functional view. Refer to D3 for the detail of mobile station.

(1) Tag type
It is simple type mobile station, which can read only proper data. It is controlled by one way access from base station.

(2) Non-Network type
It can offer the two-way non-network application access.

(3) Network type
It can offer the two-way network application access.

(4) Multi-functional type
It has multiple the ASL-NCP and can be correspondence to tag-type, non-network type, and network type

D2 DSRC-ASL Profile

This subclause shows the DSRC-ASL profile correspondence to classified mobile station and installation range. Table D2-1 shows the content of the DSRC-ASL profile of each type of mobile station, Table D2-2 shows the installation range correspondence to Table D2-1.

Table D2-1 — DSRC-ASL Profile

			Tag	Non-IP terminal	IP terminal	Multi-functional
ASL-NCP	Link Control Management		—	M	M	M
	LPCP		M(*)	M	—	M
	LANCP		—	—	O	O
	PPPCP		—	—	O	O
ASL-ELCP	Client/Server type Link Control		—	M	M	M
	Bulk transmission Control		—	O.1	O.1	O.1
	Broadcast mode processing		—	O.2	O.2	O.2
	Link Control Management	Link Connection Control	—	M	M	M
		Management Service	—	M	M	M
		Access management	—	O.3	O.3	O.3

*: base station can only registered .

Table D2-2 — Installation Range of ASL

			Tag	Non-IP terminal	IP terminal	Multi-functional
ASL-NCP Service	SendUnitData		—	M	M	M
	Echo/EchoReply		—	M	M	M
	EventInformation (Connection/Disconnecti Report)		M	M	M	M
	EventInformation（others）		—	M	M	M
ASL-ELCP Protocol	Bulk transmissionControl		—	O.1	O.1	O.1
	Broadcast Mode Cotrol		—	O.2	O.2	O.2
	Communication Control Protocol	echo	—	M	M	M
		echoReply	—	M	M	M
		eventReport	—	M	M	M
		challenge	—	O.3	O.3	O.3
		signature	—	O.3	O.3	O.3
DSRC Service	RegisterApplicationRSU/OBU		M	M	M	M
	DeregisterApplication		M	M	M	M
	NotifyApplicationRSU/OBU		M	M	M	M
	EndApplication		M	M	M	M
	NotifyApplicationVechicle _Release		M	M	M	M
	GET		—	—	—	—
	SET		—	—	—	—
	ACTION	—	—	M	M	M
		—	—	M	M	M
	Layer Management Service		—	O	O	O

NOTE M: Mandatory, O: Optional, O<n>Option: It should be installed the option which is specified at the same, number <n> in Table D2-1, —: Not Applicable

D3 Detail of Mobile Station

D3.1 Tag Type Mobile Station

D3.1.1 Overview

Tag type is simple type mobile station. It is voluntary transmitting type, which transmits automatically identifier when it detects the base station.

D3.1.2 Structure of the Tag Type Mobile Station

Figure D3-1 shows the supposed structure tag type. Figure D3-2 shows an example of transaction .It simplifies the structure of tag type using the link incomplete when the same ASL-NCP doesn't exist in the initial connection procedure of DSRC platform.

(1) ASL-ELCP
It has the only function managing the identifier, registering the profile of mobile station and disconnecting.

(2) ASL-NCP
Not implemented.

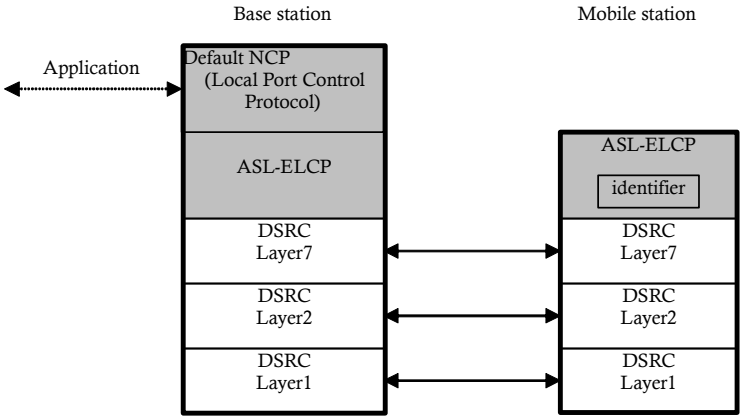


Figure D3-1 — Tag Type Structure

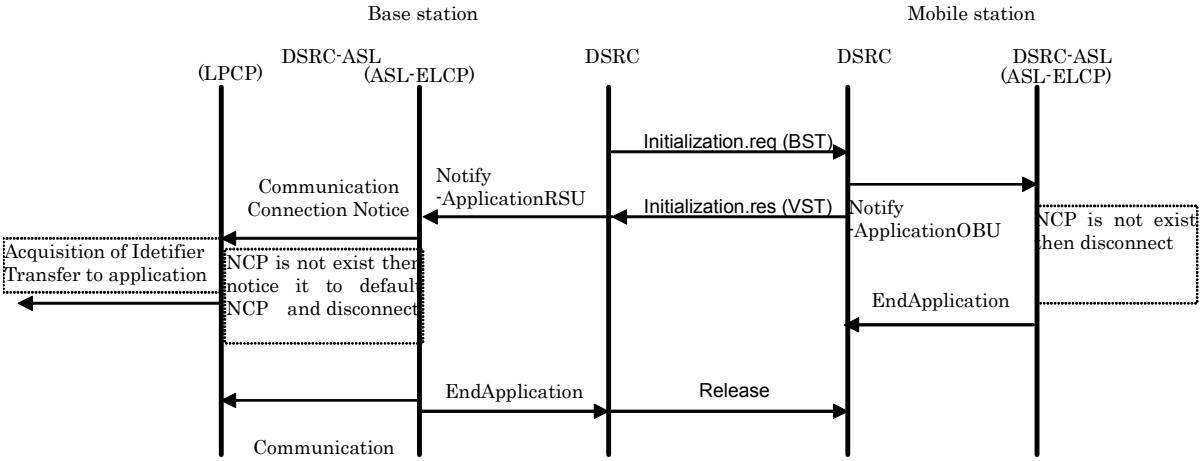


Figure D3-2 — Example of Transaction with the Tag Type Mobile Station

D3.1.3 Correspondence of Base Station with the Tag Type Mobile Station

Although mobile station doesn't load ASL-NCP for simplification, base station needs the installation of ASL-NCP for the interface to application device. Base station supporting the tag type reports connect / disconnect of link to default ASL-NCP.

D3.2 Non-Network Type Mobile Station

D3.2.1 Overview

Non-Network Type is loaded the function of non-network communication. It offers client/server type two-way communication to non-network application.

D3.2.2 Structure of the Non-Network Type Mobile Station

Figure D3-3 shows the protocol structure non-network supposed. Figure D3-4 shows an example of transaction.

(1) ASL-ELCP
It needs the minimum of function which is management of link connection and client / server type link control.

(2) ASL-NCP
LPCP

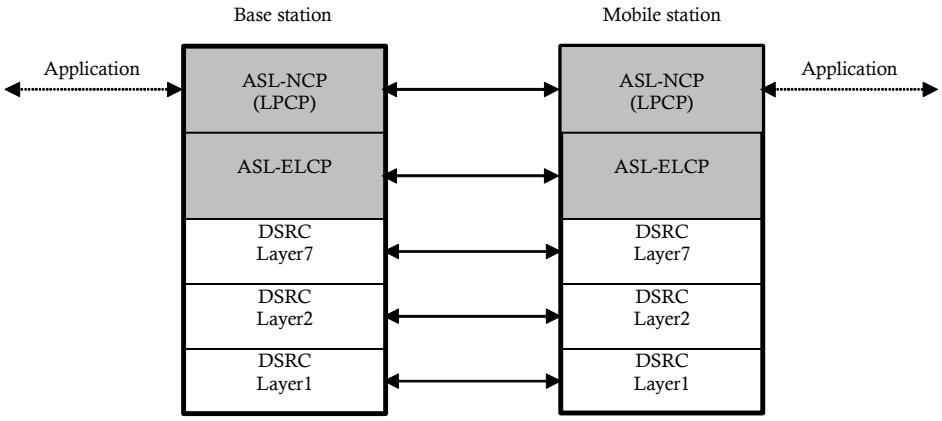


Figure D3-3 — Protocol Structure of the Non-Network Type Mobile Station

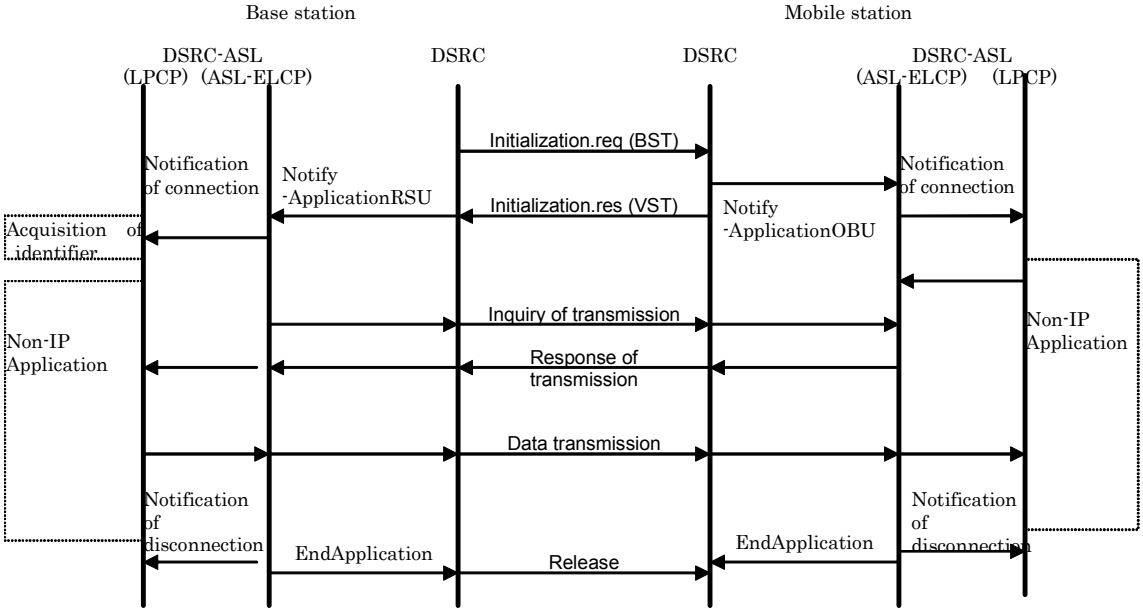


Figure D3-4 — Example of transaction to the Non-Network Type Mobile Station

D3.3 Network Type Mobile Station

D3.3.1 Overview

Network Type is loaded the function of network communication. It offers client / server type two-way communication to network application.

D3.3.2 Structure of the Network Type Mobile Station

Figure D3-5 shows the protocol structure non-network supposed. Figure D3-6 shows an example of transaction.

(1) ASL-ELCP
It needs the minimum of function which is management of link connection and client / server type link control.

(2) ASL-NCP
It needs the selection from LANCP and PPPCP for the protocol to connect internet according to its use.

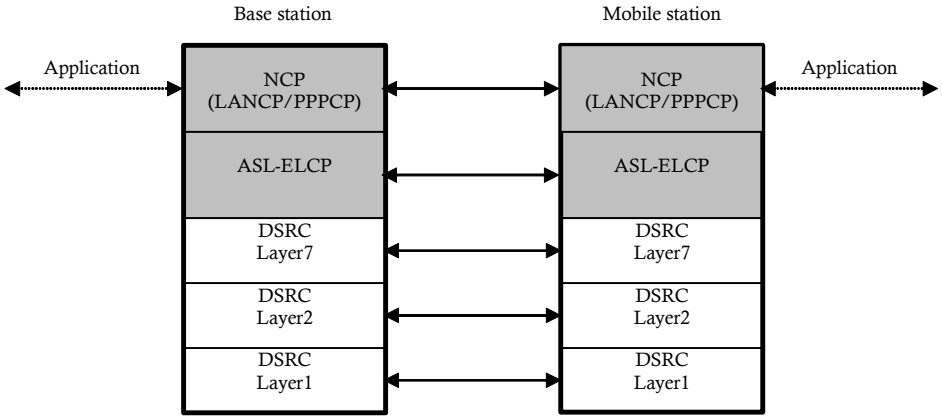


Figure D3-5 — Protocol Structure of the Network Type Mobile Station

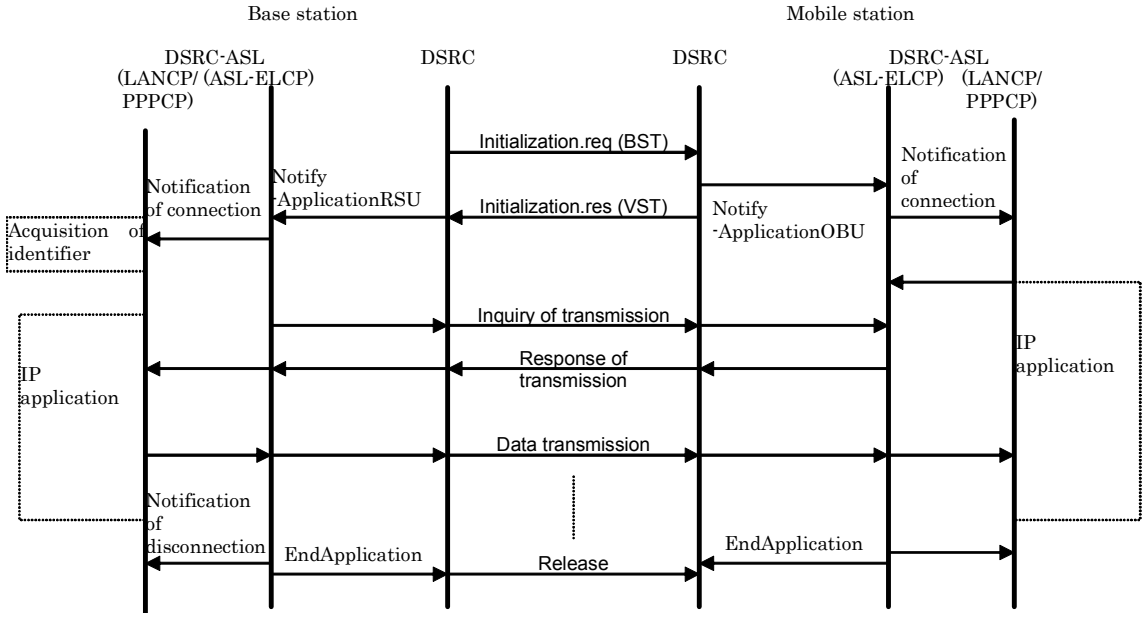


Figure D3-6 — Example of Transaction to the Network Type Mobile Station

D3.4 Multi-functional Type Mobile Station

D3.4.1 Overview

Multi-functional Type is loaded multiple ASL-NCP and correspondence to tag-type,

non-network type, and network type.

D3.4.2 Structure of Multi-functional Type Mobile Station

Figure D3-7 shows the protocol structure multi-functional type supposed. Figure D3-8 shows a example of transaction.

(1) ASL-ELCP
It needs the minimum of function which is management of link connection and client / server type link control.

(2) ASL-NCP
It needs the selection according to its use.

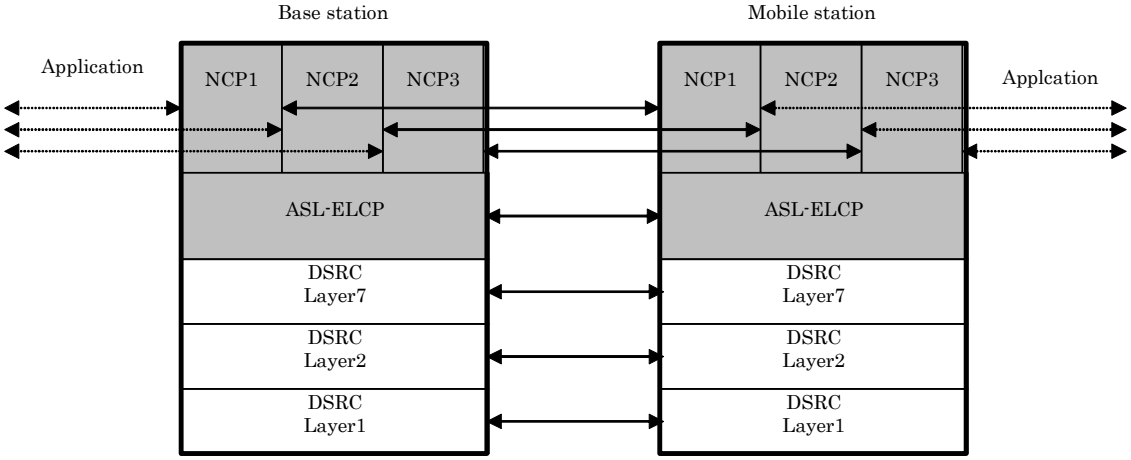


Figure D3-7 — Protocol Structure of the Multi-functional Type Mobile Station

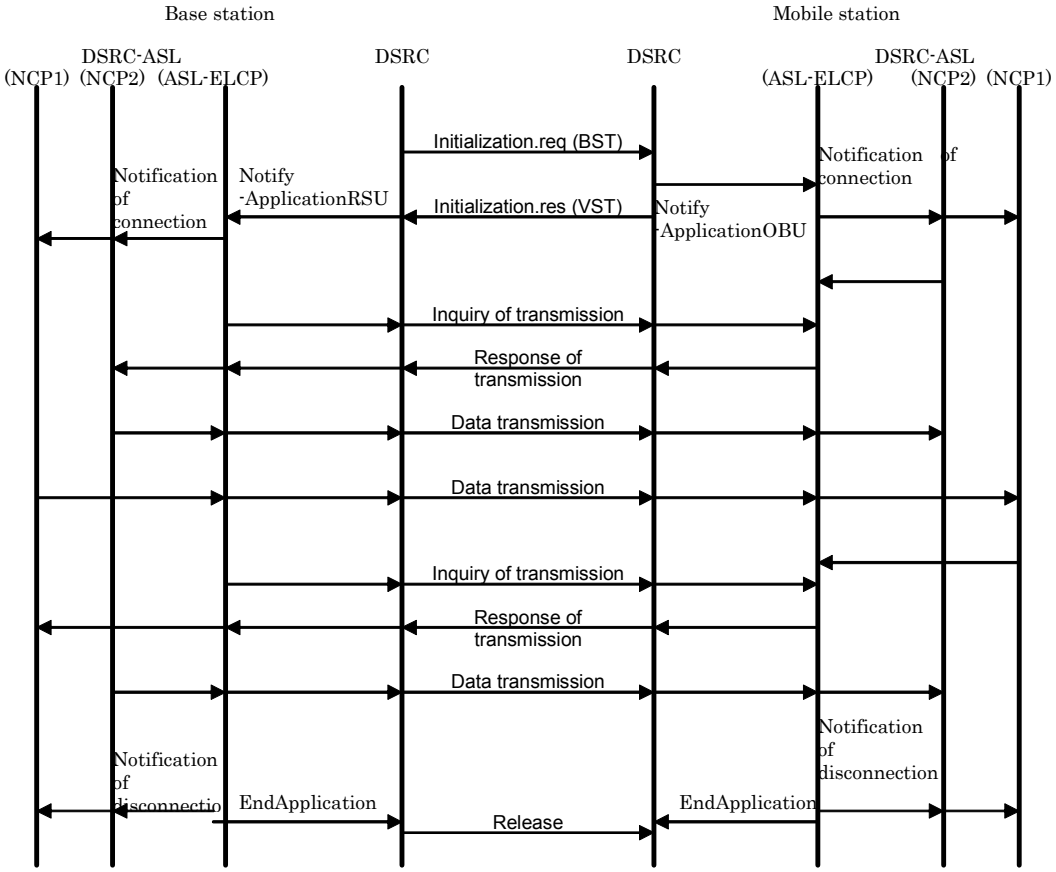


Figure D3-8 — Example of Transaction to the Multi-functional Type Mobile Station

Annex E
(normative)
Version Management

E1 Version of DSRC Application Sub-Layer (DSRC-ASL)

E1.1 Definition of Version

The version of the DSRC-ASL shall show the edition of protocol specification in an ASL-ELCP and an ASL-NCP.

E1.2 Purpose

The version of the DSRC-ASL is used to select the applied protocol of the DSRC-ASL in the session between base station and mobile station. The base station and the mobile station should show the version of the implemented DSRC-ASL feature by a version index of the DSRC-ASL profile.

E1.3 Numbering

The version index is integer from “0” to “15”, the first edition is defined as “0”. When it is revised, the edition is added “1” to before revised edition.

E2 Revision

The revision is done for the change of function component of prescribed extended link control protocol and protocol of prescribed ASL-NCP as a principle.

NOTE It is supposed that the addition of function component of the prescribed ASL-ELCP and ASL-NCP are solved in set up procedure.

In the case of the revision of the DSRC-ASL, it shall make a revision draft paying attention to the following:

(1) Both the base station and the mobile station have the effect of link to use renewal version.
(2) It should be considered that the revision does not interfere with the operation of the former version.
(3) The DSRC-ASL of renewal version should be implemented the former version DSRC-ASL and assure the communication using the former version within permission of functional limit.

We should ask the mobile station to carry out the renewal of version smoothly keeping the above-mentioned consideration that variables, data field etc. treated as reserved in former version is not treated as supposed specific value in installation recognizing for future

extension.

E3 Selection of Protocol

E3.1 Version Information Setting

It is supposed that the protocol between the base station and the mobile station which are different version should be selected the version which both the base station and the mobile station are available. The base station and the mobile station should select the newest version and set up the DSRC-ASL profile when there are some available versions.

E3.2 Selection Procedure

The following is procedure of protocol by version index.

(1) The base station and mobile station confirm the version index from received DSRC-ASL profile.
(2) When the received version index coincides loaded DSRC-ASL version, the base station and the mobile station carry out the each protocol. (Figure E3-1)
(3) When the received version index does not coincide the loaded DSRC-ASL version, the base station and the mobile station compare each version and select lower version.
(4) The base station and the mobile station, which have selected version, carry out the protocol. (Figure E3-2)
(5) The base station, which has no selected version, does not move the protocol and ends the link after issuing “EndApplication” primitive. (Figure E3-2)
(6) The mobile station, which has no selected version, does not move the protocol and transmits the event report message “Not available designated version”, and ends the link after issuing “EndApplication” primitive. (Figure E3-3)

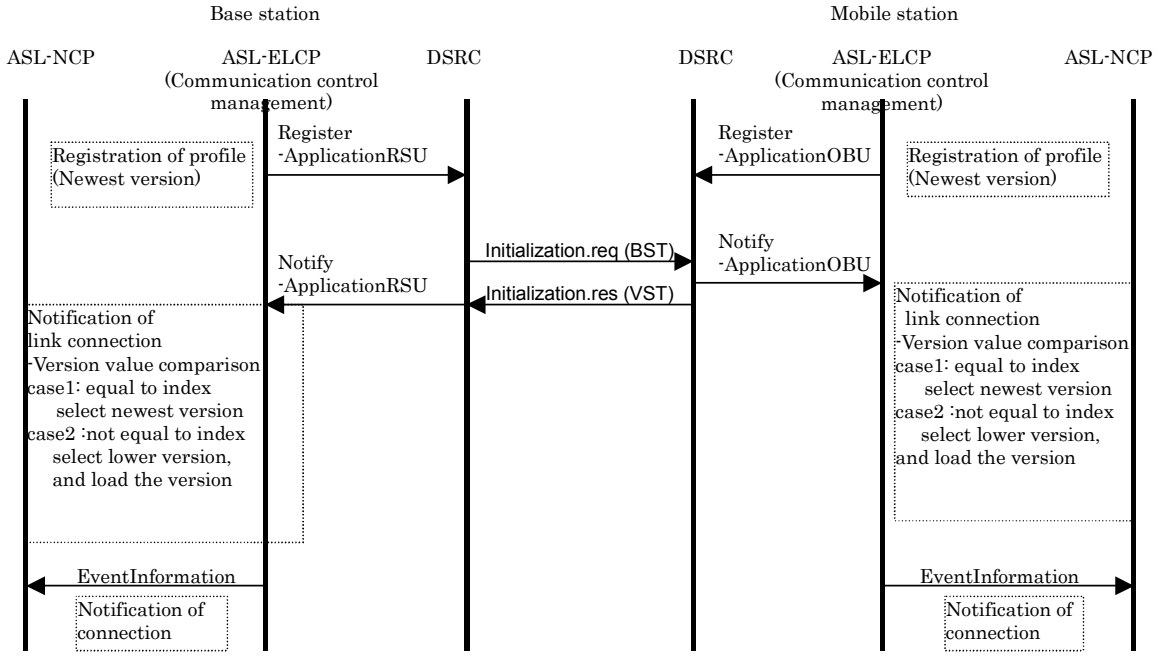


Figure E3-1 — Version Selection Procedure (1)

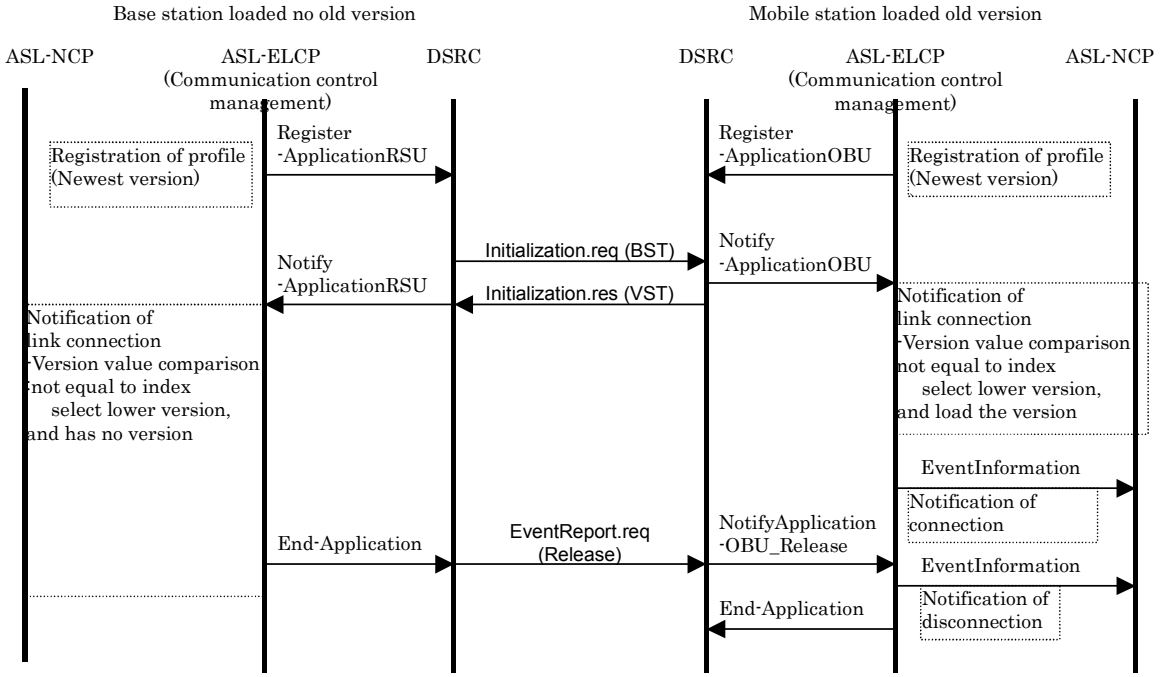


Figure E3-2 — Version Selection Procedure (2)

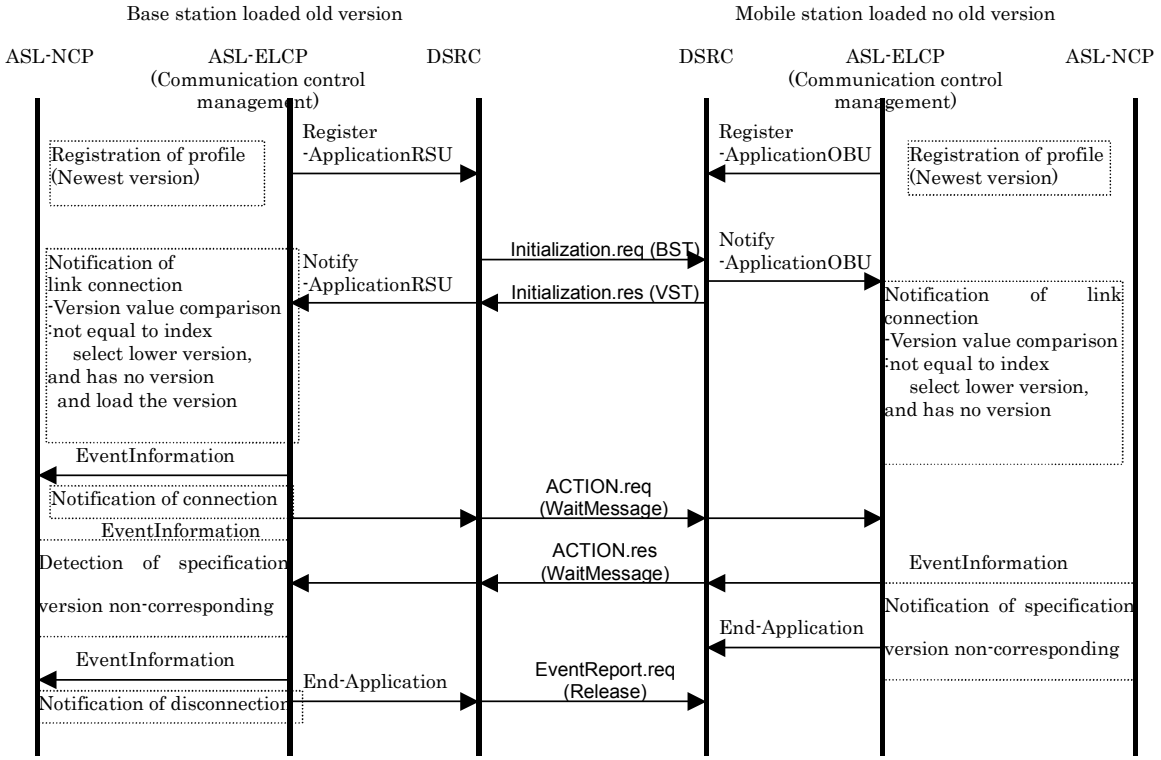


Figure E3-3 — Version Selection Procedure (3)

Annex F
(normative)
Mobile station Identifier

F1 Definition

A Mobile station Identifier is defined inherent number, which can identify mobile station.

F2 Numbering

Mobile station Identifier uses a unique identification number, which is original data to generate Wireless Call Number Channel specified in the ARIB STD-T75. It shall show the following configuration.

Octet	Bit (MSB) 7	6	5	4	3	2	1	(LSB) 0
1	a47	a46	a45	a44	a43	a42	a41	a40
2	a39	a38	a37	a36	a35	a34	a33	a32
3	a31	a30	a29	a28	a27	a26	a25	a24
4	a23	a22	a21	a20	a19	a18	a17	a16
5	a15	a14	a13	a12	a11	a10	a9	a8
6	a7	a6	a5	a4	a3	a2	a1	a0

From “a0” through “a47”, they are corresponded to each digit from 1st through 48th of ID name, which is converted from number of 12 digits into BCD code.

F3 Registration

The registration of identifier to mobile station should compliant with the registration of the WCNC at the same time. In this case the result of coding the mobile station identifier as rule of WCNC coding should coincide Identity number (IDNR) of WCNC.

See Annex C in ARIB STD-T75 about the rule of coding to IDNR.

Annex G
(informative)
DSRC-ASL Extended Link Control Protocol (ASL-ELCP)

G1 Bulk Transmission

G1.1 Outline

Bulk transmission improves effective transmission rate by assigning several slot in the DSRC transmission frame to the same mobile station. In the layer 2 protocol of ARIB STD-T75, it is specified that the transmission frame isn't occupied by a specific mobile station, it can occupy several slot in a frame by the upper layer. Using this specification can realize the bulk transmission.

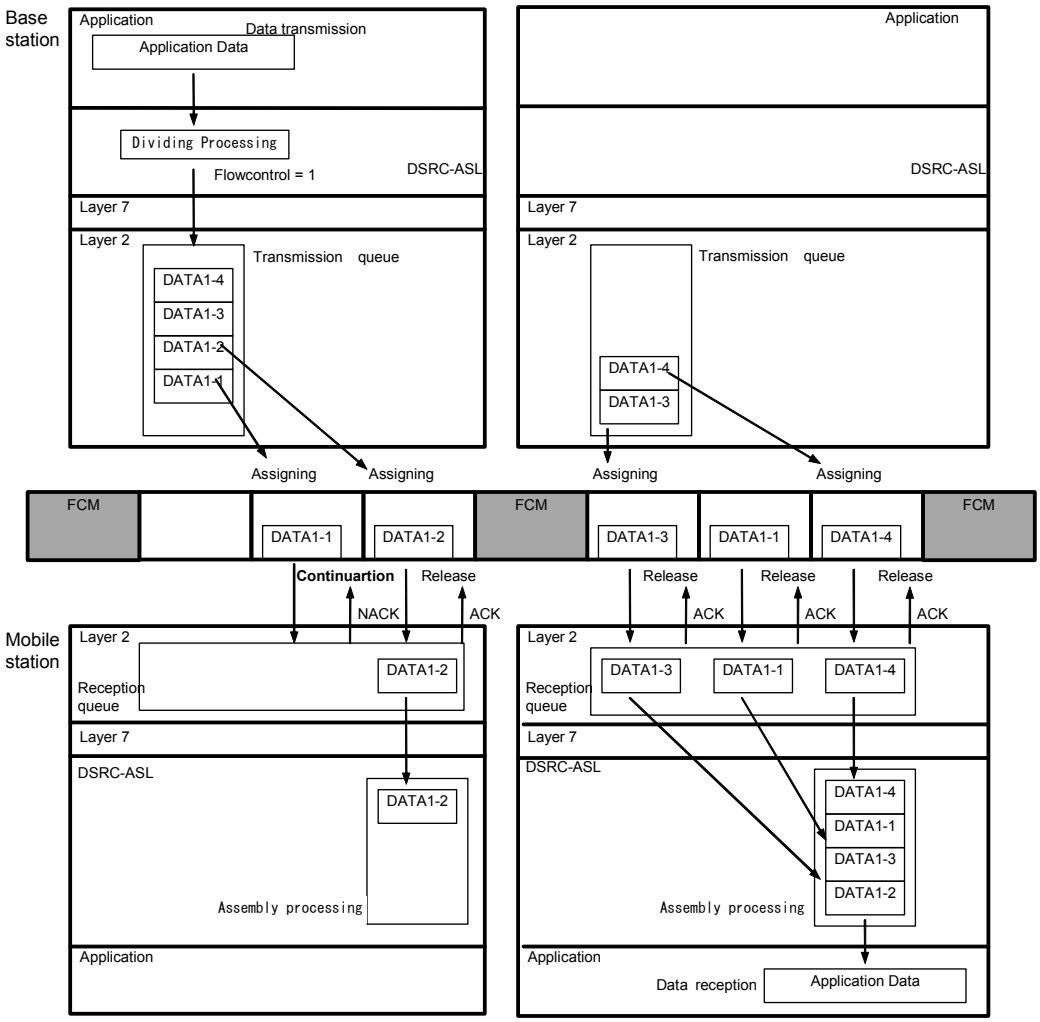


Figure G1-1 — Outline of Bulk Transmission Operation

Figure G1-1 shows that the outline of bulk transmission operation. DSRC-ASL operates dividing SDU to the size, which can be housed in a slot of transmission frame when receiving the bigger size of SDU from application and passing to layer 7 in turn. In Figure G1-1, the packet divided by this operation (bulk segment) is stored in transmission queue and assigned to open slot in turn.

In receiving operation, the bulk segment of assigned slot is taken and stored in the receive queue of DSRC-ASL. In this case, it is supposed that re-send transaction operates for transmission error in the radio section, as it isn't assured that bulk segment is stored in turn in the receive queue, it constructs the origin SDU distinguishing the order sequence number added on bulk segment in receiving operation.

G1.2 Slot Assignment

Table G1-1 shows example of assignment pattern assigning the bulk segment on the transmission frame of DSRC. And Figure G1-2 shows outline of operation carrying out the stripe assignment in the example. In this figure base station carries out OBE A and B, they also assign the packet enable to receive twice rate of up link.

Table G1-1 — Example of Slot Assignment of Bulk Segment

	Slot Assignment pattern	Outline
1	Block Assignment type |AAAABB|BBCCCC|	Block-sizing bulk segment in a mobile station unit.
2	Block Assignment type in frame |AABBCC|AABBCC|	Dividing the frame by the number of mobile station, and block-sizing bulk segment.
3	Stripe type |ABCABC|ABCABC|	Assigning the slot as a bulk segment unit.

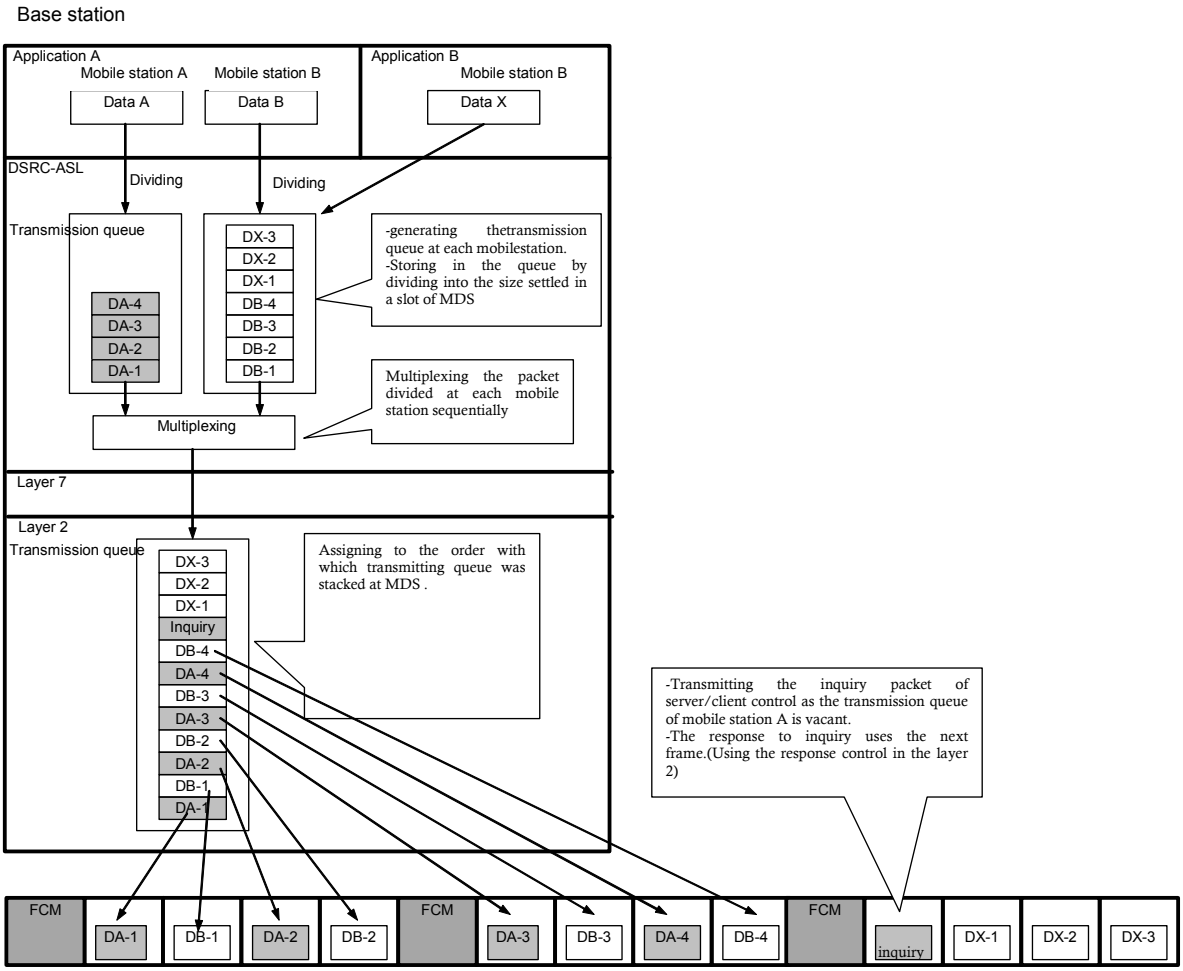


Figure G1-2 — Bulk Transmission by Stripe Assignment

G1.3 Band Control

As the bulk transmission is best effort type service which divides communication band by the number of mobile station, the effective transmission rate falls less than the rate of 1slot when number of mobile station exceed number of slots which compose frame.

There the following method in the case it offers the QoS service which assures the lower limit of communication band to restrain the effective transmission rate falling less than rate of 1 slot.

(1) Band restriction type control
When the number of mobile station connected link is less than the number of slots, which compose frame, it applies bulk transmission control independently of data size.

When the number of mobile station connected link excess the number of slots, which compose frame, it doesn't apply bulk transmission control.

(2) Band reservation type control
When the number of mobile station connected link exceeds the number of slots, which compose frame, it manages to reserve the bulk transmission control.

In Meet type band control, a new slot queue equal to the number of communication slot in a frame is generated, the number of multiplex is set up not to exceed the number of slot applying multiplexing the slot queue.

It can perform to add the operation, which is enclosed dot line in Figure G1-3. The operation of bulk segmentation can be performed in the process dividing.

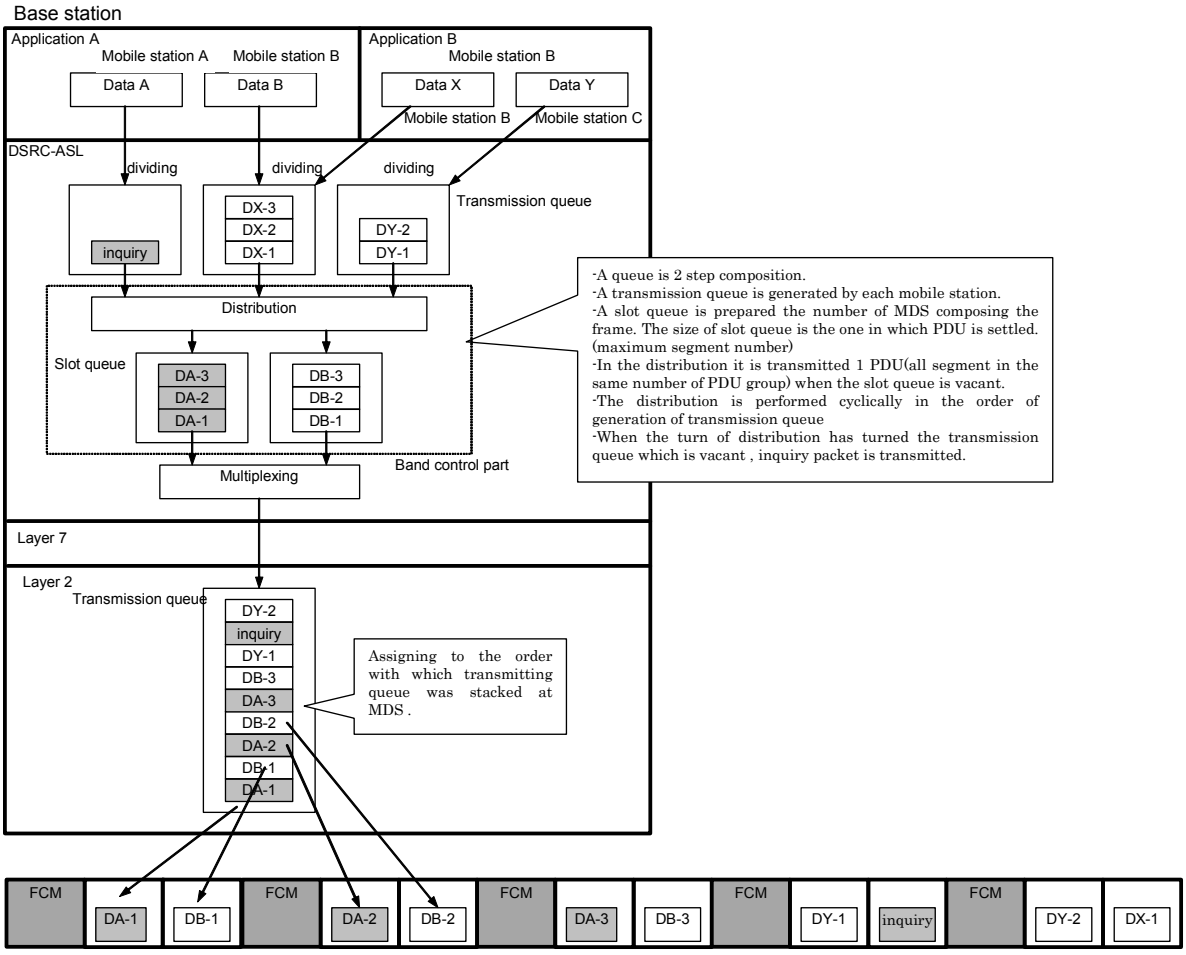


Figure G1-3 — Band Reservation Type Control

G2 Broadcast Mode

G2.1 Outline

In Broadcast mode data is delivered k times over for the general public mobile station using the broadcast link address provided by DSRC layer 2. Figure G2-1 shows outline of

broadcast mode processing operation. When base station receives SDU delivered in broadcast mode from ASL-NCP, it contains in transmission queue bulk-segmenting according to be received. The transmission queue for broadcast, which is different from the one for bulk transmission control transmits the data N times repetitions to improve the bit error rate. (Continuation transmitting processing)

The mobile station detects the broadcast link address and assigns received data to broadcast mode processing. The reception data assigned by broadcast mode processing constructs the bulk segment similar to bulk transmission control. In this process missed bulk segment utilizes later arrived bulk segment. When the same bulk segment is received after finishing construction, they are all cancelled.

In the construction processing, it is cut off in valid time provided by each transmission process, and the received queue should be cancelled after cut off.

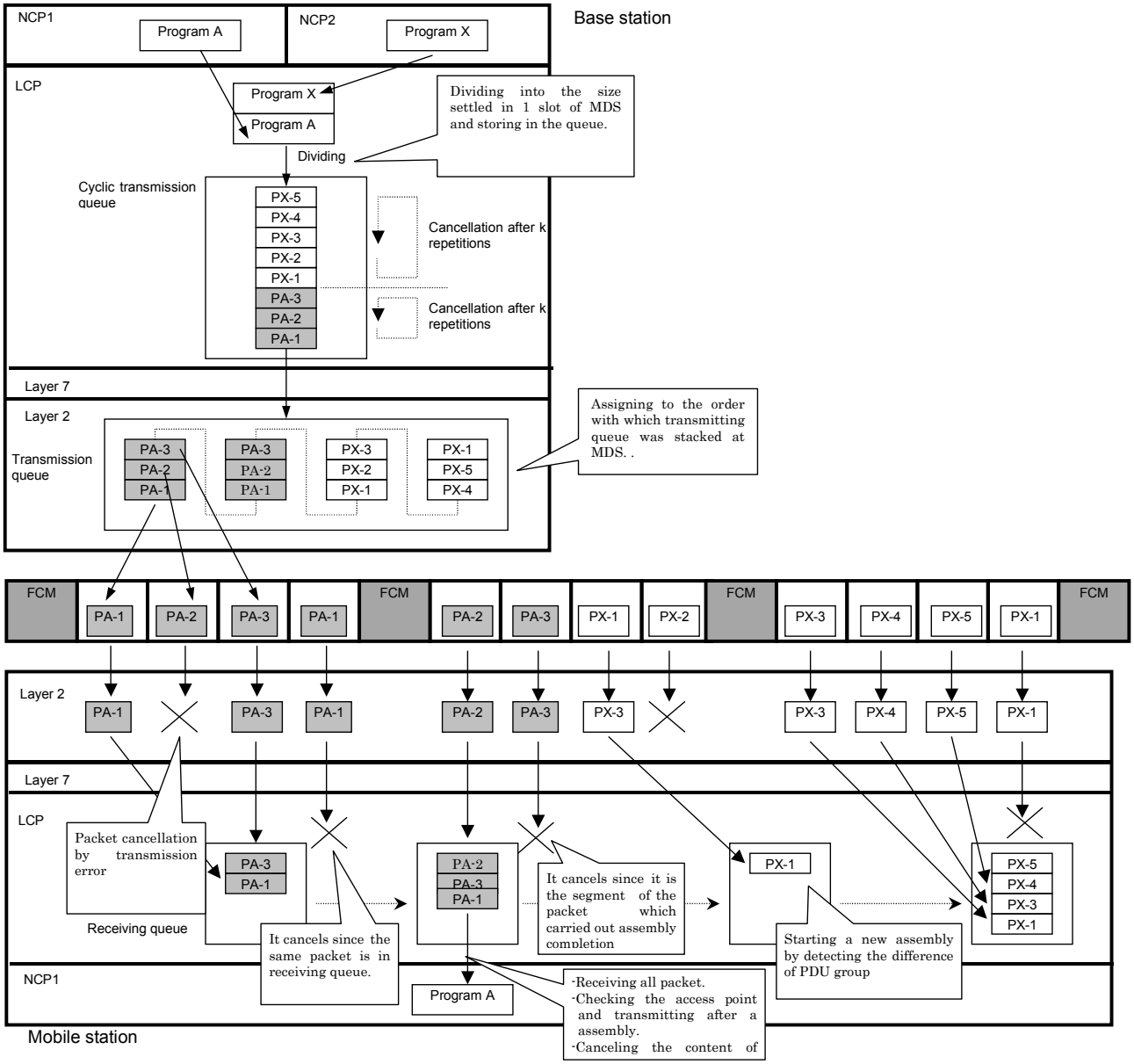


Figure G2-1 — Outline of Broadcast Mode Processing Operation

G2.2 Repetition Transmitting Process and Bit Error Rate in Broadcast Mode Process

Figure G2-2 shows receiving error rate in ASK mode regulated P_r ARIB STD-T75.

The error rate of data in broadcast mode P_r equals to probability which a Message Data Channel (MDC) composed DSRC transmission frame becomes wrong when broadcast data from ASL-NCP is divided into m bulk segments and transmitted assigned DSRC transmission frame, and is calculated from equation G2-1. The receiving error rate P of MDC is calculated as the probability which Frame Control Message Channel (FCMC) or MDC

becomes wrong. (Which subtracts the probability Pf*Pm, which both FCMC and MDC become wrong from the sum of FCMC packet error rate Pf and MDC packet error rate)

$$Pf = 1 - (1 - Pb)^{FCMC} \qquad \text{------(equation G2-1)}$$

$$Pm = 1 - (1 - Pb)^{MDC}$$

P : error rate of MDC
Pf : packet error rate of FCMC
Pm : packet error rate of MDC
Pb : bit error rate of link (=1*10 E -5)
FCMC: packet length of FCMC (464bit = 58*8)
MDC: packet length of MDC (568bit = 71*8)
m :number of slot(MDS) which divided ASL-SDU contains

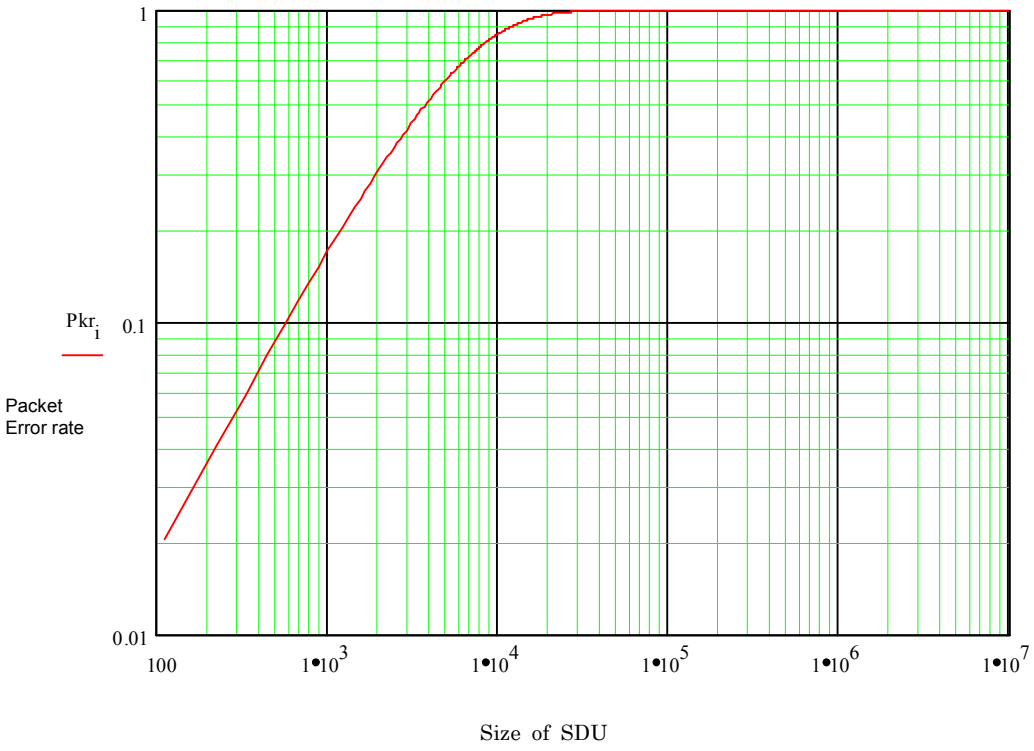


Figure G2-2 — Receiving Error Rate in Broadcast Mode Process (ASK mode)

In broadcast mode without re-try function like point to point communication receiving rate becomes about 80% when the size of ASL-SDU is about 1kB and is impossible to receive when the one becomes more than 10kB. So, Continuation transmitting processing which can receive the same packet at several times in communication area as alternative to retry is required to secure reliability of system.

Next, we calculate the required repetition frequency (minimum repetition frequency which mobile station can receive) as base station offers the service, which is secured quality of

service.

As the packet error rate with transmission k times repetition can be calculated from the probability which all k times becomes wrong, broadcast data error rate can be calculated from equation G2-2 with transmission repeat. (As in the broadcast mode the probability, which all k times become wrong is equal to the probability with transmission repeat.

Figure G2-3 shows result of improvement by continuation transmitting process.

$$P_k = P^k \qquad \text{------(equation G2-2)}$$

$$P_r = 1 - (1 - P_k)^m$$

P_k : packet error rate with transmission repeat
P : error rate of MDC(see equation G2-1)
k : transmission repetition times
m : number of slot(MDS) which divided ASL-SDU contains

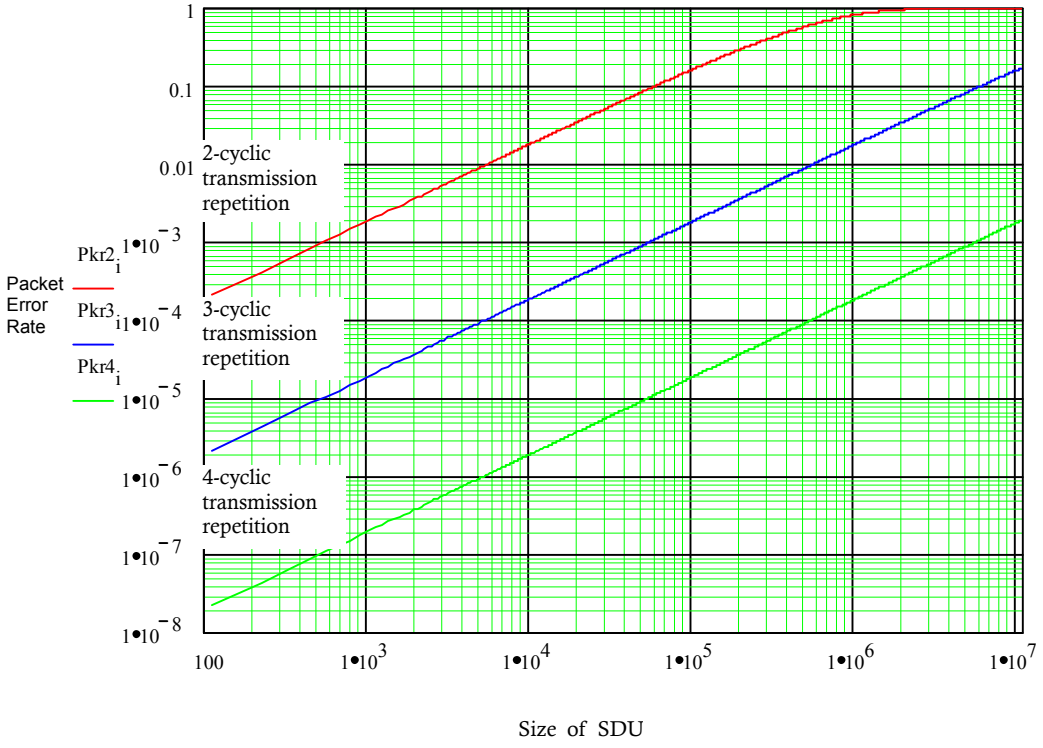


Figure G2-3 — improvement of Error Rate by Repetition transmitting Process (ASK mode)

G2.3 Hybrid processing with the Point to Point Communication

Figure G2-4 shows an example of mixture processing with the point to point communication.

(1) Assignment of slot

In this example it is multiplexed stripe assignment type as counting a mobile station in multiplexed at block assignment type in the case of mixture processing with the point to point communication.

In this case the slot for broadcast mode should be secured more than one slot in a frame. And the volume of data offered in broadcast mode should be calculated as the volume of a slot data in a frame.

(2) Band control of bulk transmission at the hybrid processing type
As the slot for broadcast mode should be secured more than one slot in a frame, the number of slot should be border condition as the one of Message Data Slot (MDS) composed link frame minus 1 in the case of mixture processing with cut off type band control.

And transmission queue should be the one of MDS composed link frame minus 1 in the case of mixture processing with meet type band control. (See Figure G2-4)

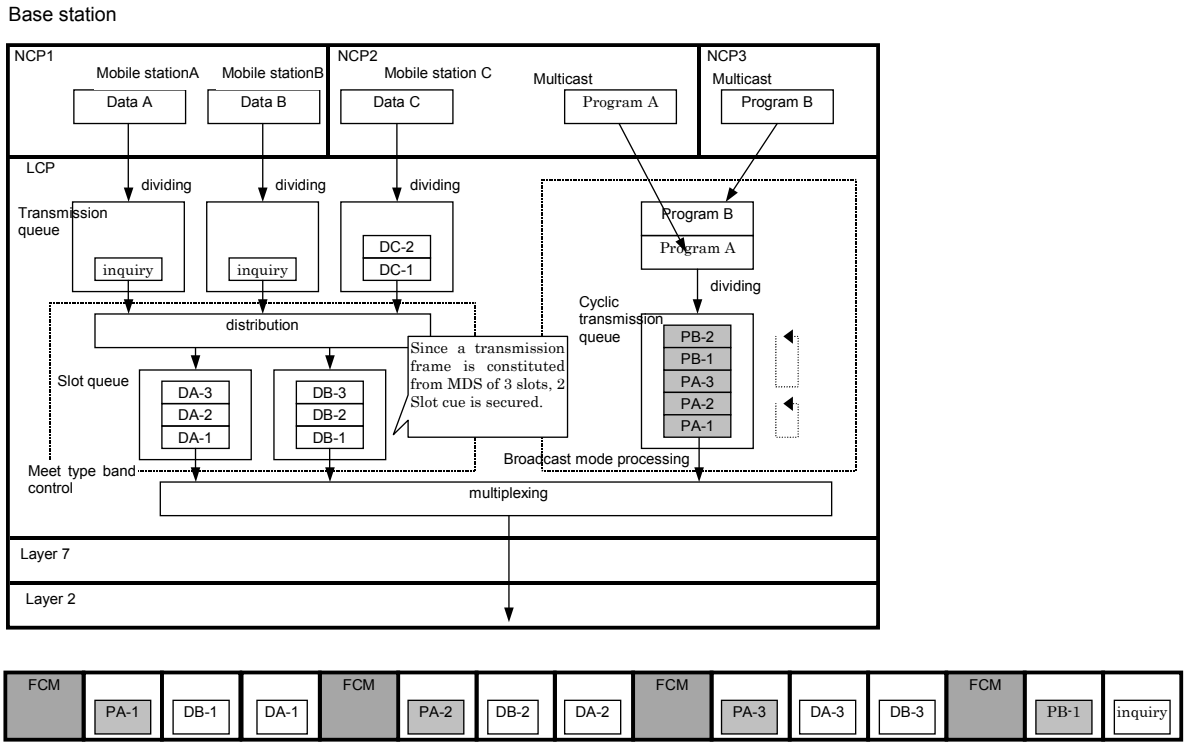


Figure G2-4 — Band Reservation Type Control of Bulk Transmission at the Hybrid Process Type Base Station

Annex H
(normative)
Local Port Control Protocol (LPCP)

H1 Equipment Configuration

The LPCP specifications are for an integrated-type configuration (in which all software including non-network type applications are implemented in the base station/mobile stations). The separated-type configuration for the base station is specified as LPPoverUDP.

H2 Relationship between Service Classification and LPCP Implementation scope (Informative)

H2.1 Service Classification

Services provided using the LPCP are classified into the following two types:

(1) Broadcast type service
Broadcast service is performed from the base station to mobile stations using the broadcast mode control function of the ASL-ELCP. As mobile stations, non-network type terminals defined in Annex D1 are assumed.

(2) Bidirectional type service
This service performs individual bidirectional communication using the client/server type communication control function of the ASL-ELCP. As mobile stations, non-network type terminals defined in Annex D1 are assumed.

H2.2 Implementation Scope

Table H2-1 shows the implementation scope of each function of the LPCP corresponding to the classified services.

Table H2-1 — Implementation Scope of Local Port Control Protocol

		Broadcast type service	Bidirectional type service
Local port No.	Default NCP (0x0801)	-	M(base)
	Echo (0x0802)	-	M
	Other ports	NOTE	NOTE
LPCP function	TransferData	M	M
	EventReport (connection, disconnection and accept port list)	-	M
	EventReport (other than above)	O	O
	OpenPort	M (mobile)	M
	ClosePort	M (mobile)	M

NOTE Other port numbers are opened in accordance with mounted applications.

Symbol Description

M	Mandatory
M (mobile/base)	Mandatory only in mobile/base station
O	Optional
-	Not applicable

H3 Local Port Numbers

Local port numbers are used as the connection identifier in non-network type applications. In this specification, local port numbers are specified as shown in the table below. “1” to “0x0FFF” are reserved port numbers, and “0x1000” to “0xFFFF” are private port numbers.

Table H3-1 — Local Port Number Classification

Port No.	Application	Remarks
0	Unused	
1 to 0x07FF	Same as definition of UDP port number	RFC3232
0x0800	Unused	
0x0801	Default NCP	
0x0802	Echo application for LPCP wrap test	
0x0803 to 0x0FEE	Ports for server/peer-to-peer application	
0x0FEF	Echo application for LPP wrap test	
0x0FF0 to 0x0FFE	Application ports for test/trial system	[0x0FF0] to [0x0FF8] are used in compatibility confirmation test for mobile stations.
0x0FFF	Local port management entity	

H3.1 Relationship between Applications and Local Port Numbers

The application models are supposed to be the client/server model and peer-to-peer model. In the client/server model, reserved port numbers are used for server processes, and private local port numbers are used for client processes in general. In the peer-to-peer model, reserved port numbers are used in bidirectional processes in general. Reserved port numbers shall not be used without the management of port numbers.

When any local port other than local ports with reserved port numbers are used as receiving ports in the server processes or peer-to-peer model, local ports duplicate use problem may be caused. If erroneous connection occurs due to overlapped use of a local port, the application receiving erroneous data shall set the corresponding local port as a rejected reception port.

H3.2 Management of Local port Numbers

Although assignment and management of reserved port numbers are outside the range of this standard, consideration for management such as registration is required in using them.

H3.3 Setting of Local Port Numbers

(1) A reserved port number shall be assigned to an application without overlap globally.
(2) An application can have two or more local ports.
(3) Each application shall use receiving port numbers, which do not overlap within each station.
(4) When determination of the sending source is unnecessary or when the sending source is known, the sending source port can be omitted.

H3.4 Process Procedure Related to Local Port Numbers (Informative)

As examples of processing procedure related to the local port numbers, the processing procedure for peer to peer communication type service (client/server model), peer to peer communication type service (peer-to-peer model), and broadcast type service are described below.

H3.4.1 Peer to Peer Communication Type Service (Client/Server Model) Process Procedure

Figure H3-1 to Figure H3-6 show echo application procedure as examples of processing procedure in the client/server model.

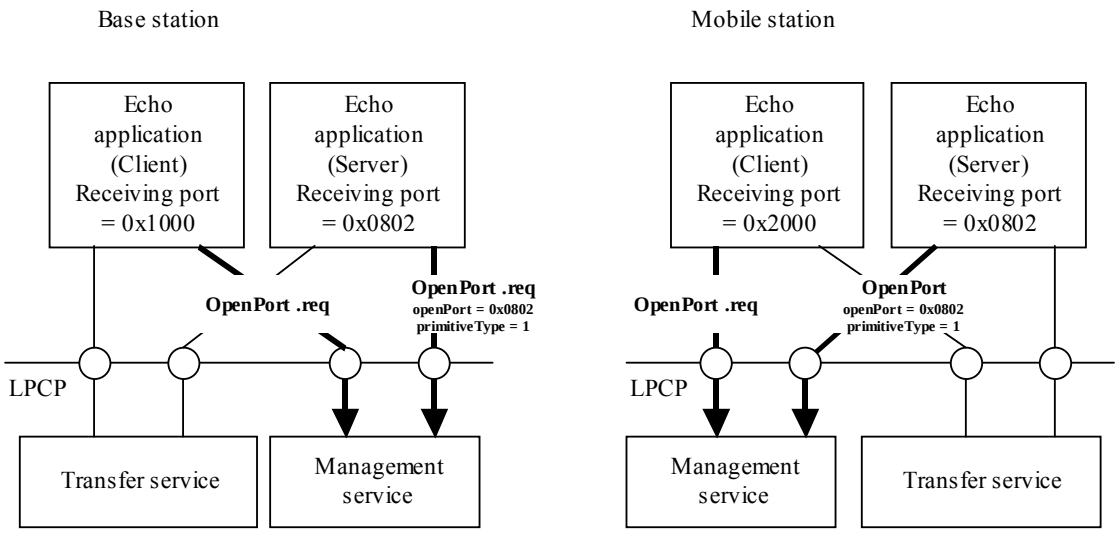


Figure H3-1 — Example of Application Registration Operation (1)

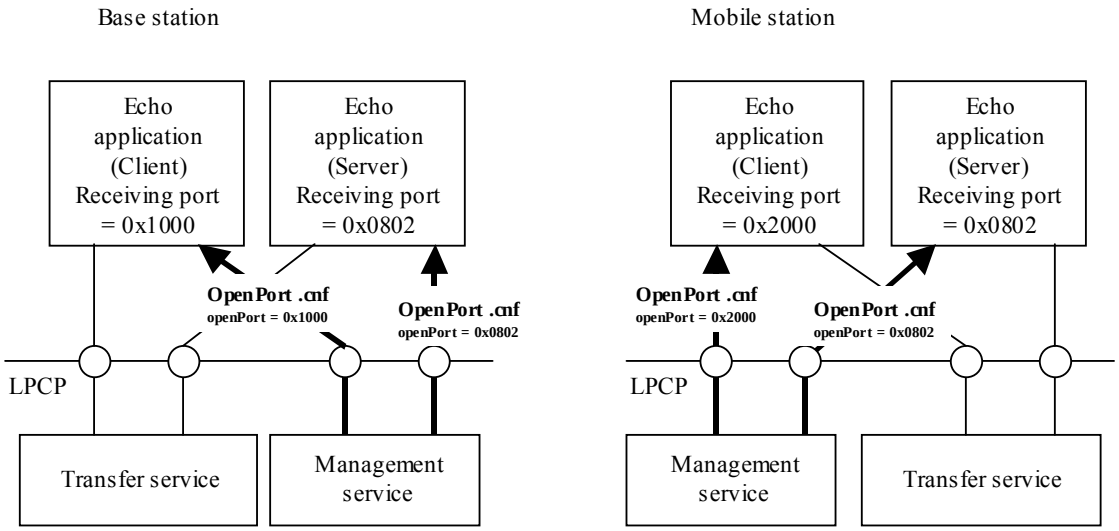


Figure H3-2 — Example of Application Registration Operation (2)

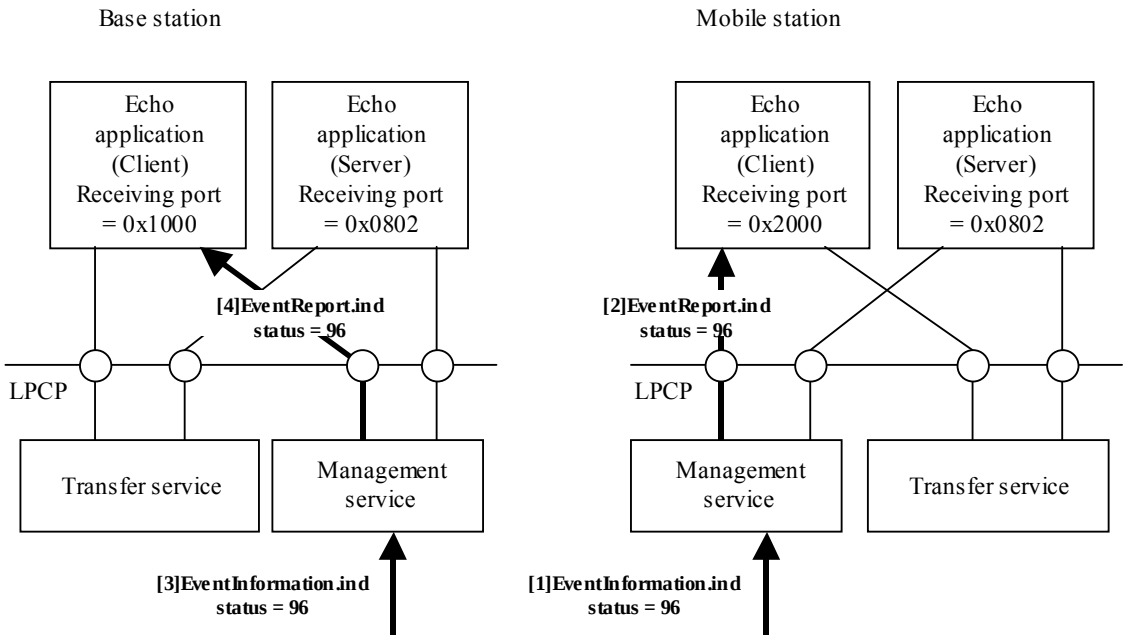


Figure H3-3 — Initial Set up Operation (connection notice)

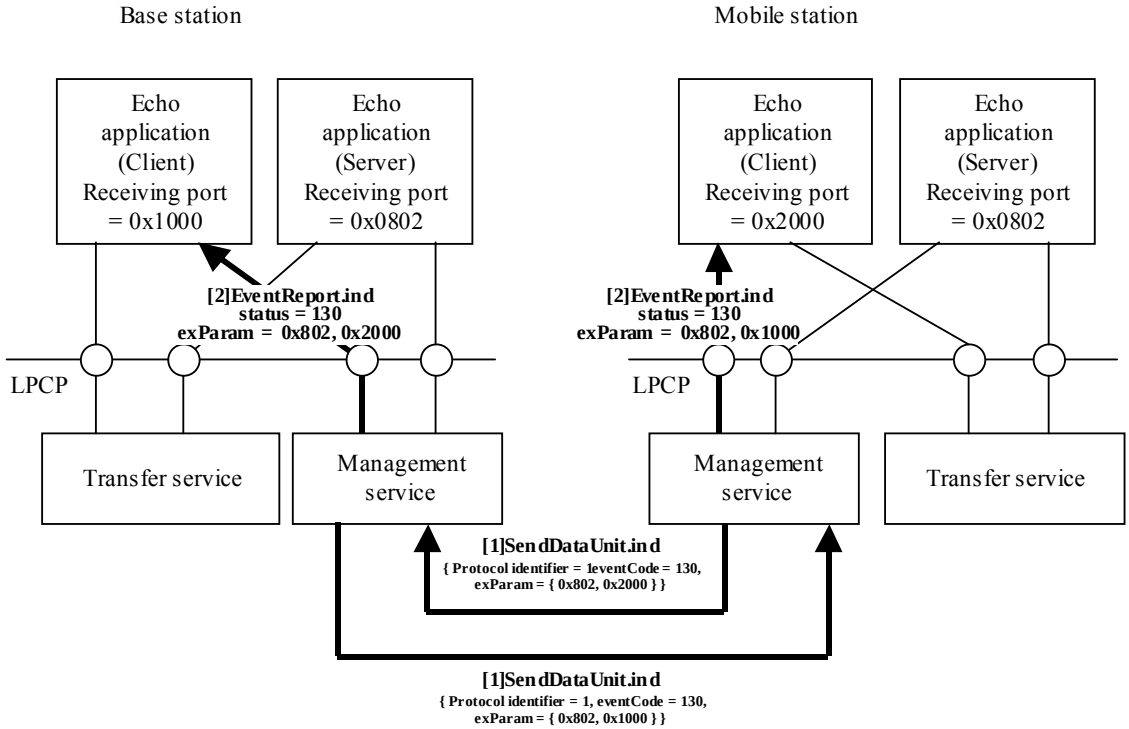


Figure H3-4 — Initial Set up Operation (accept port list exchange)

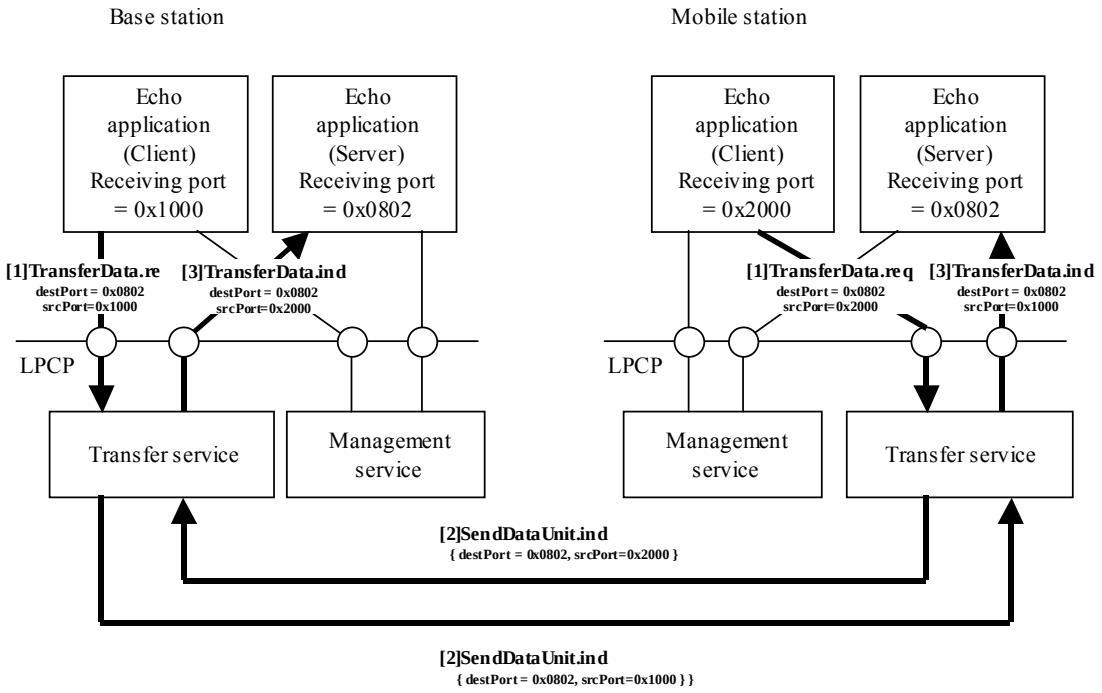


Figure H3-5 — Example of Data Transfer Operation (echo request)

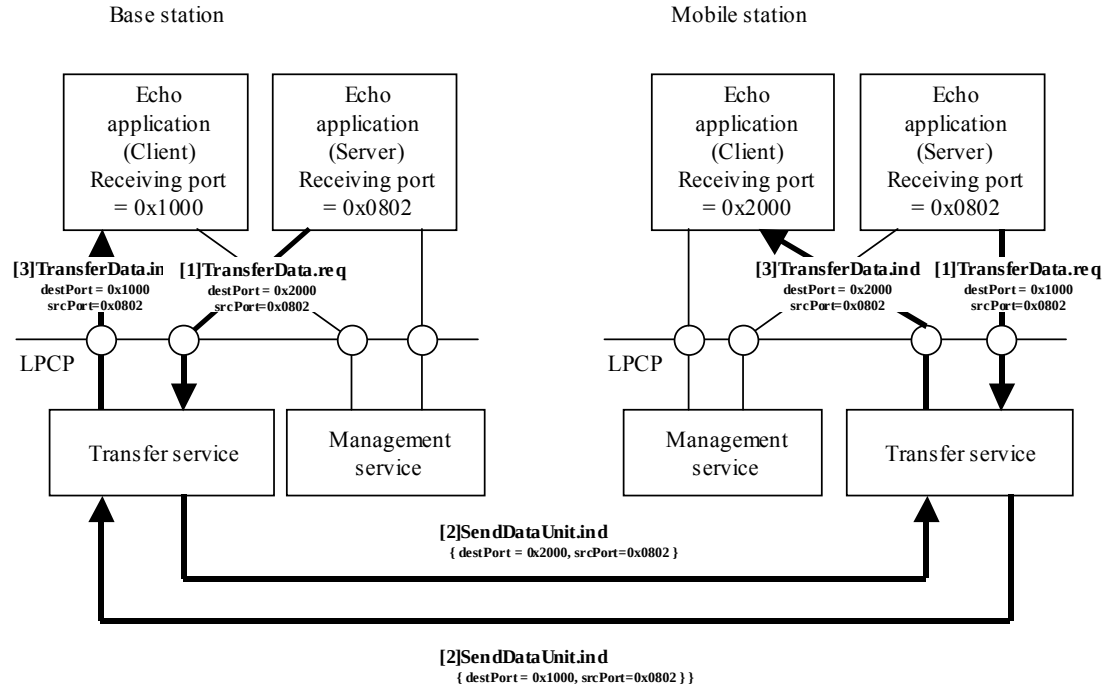


Figure H3-6 — Example of Data Transfer Operation (echo response)

H3.4.2 Peer to Peer Communication Type Service (Peer-to-Peer Model) Process Procedure

Figure H3-7 to H3-12 show examples of process procedure in the peer-to-peer model.

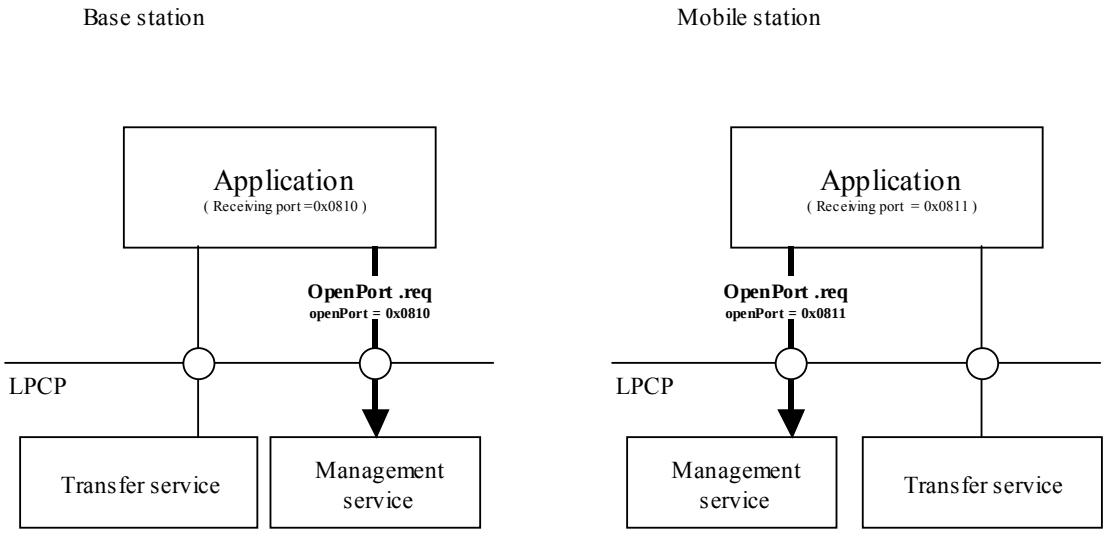


Figure H3-7 — Example of Application Registration Operation (1)

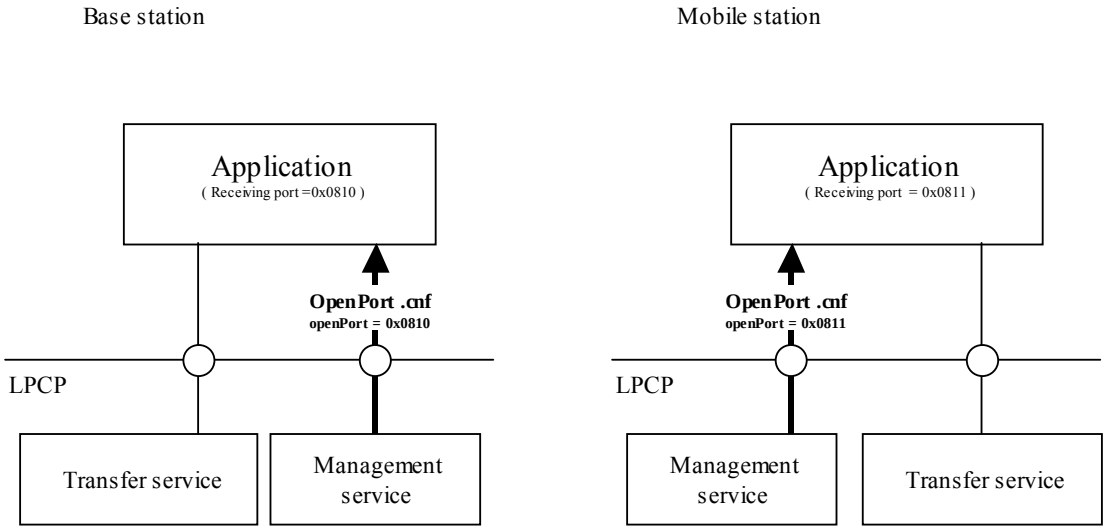


Figure H3-8 — Example of Application Registration Operation (2)

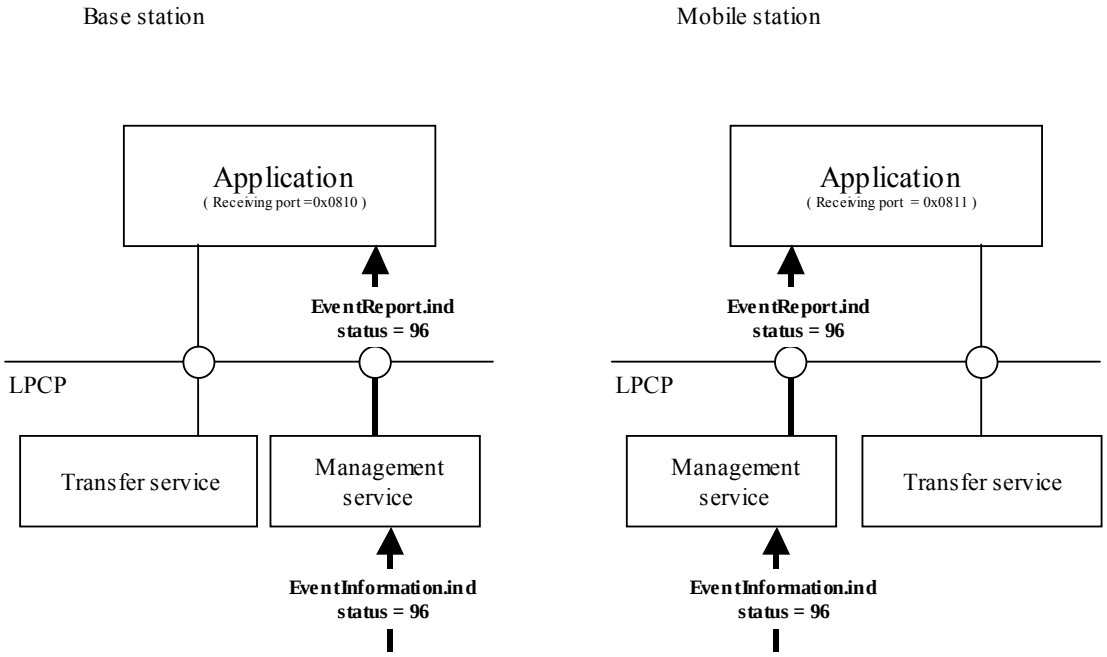


Figure H3-9 — Initial Set up Operation (connection notice)

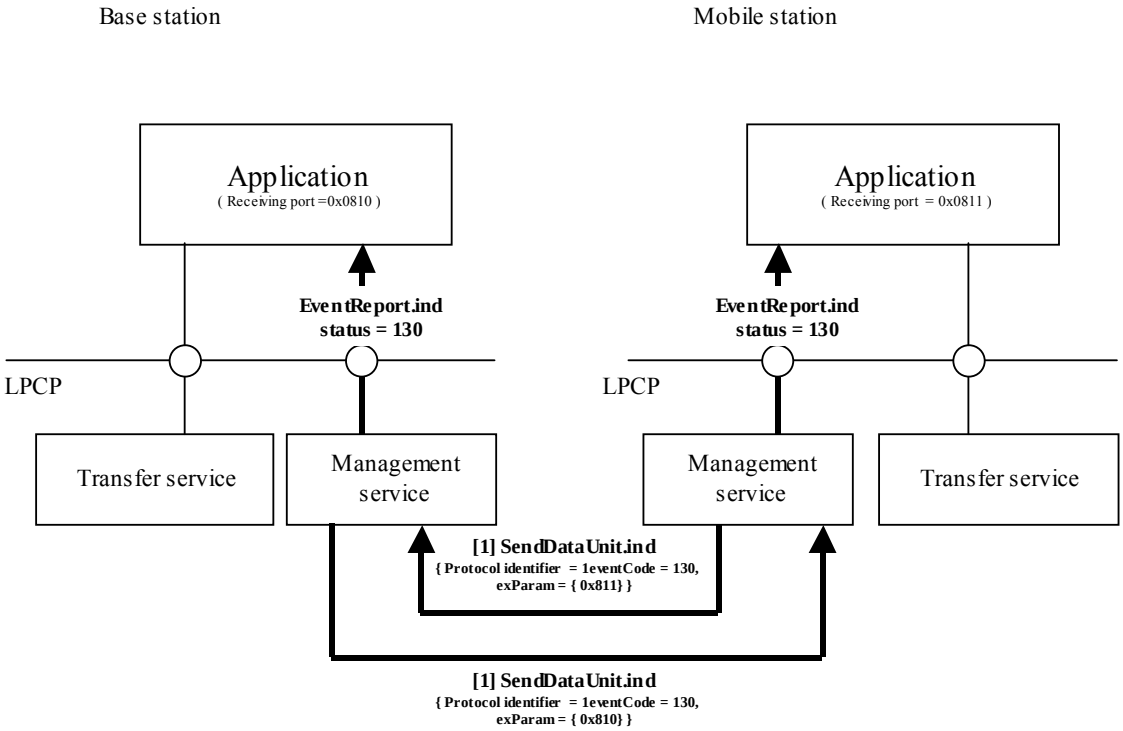


Figure H3-10 — Initial Set up Operation (accept port list exchange)

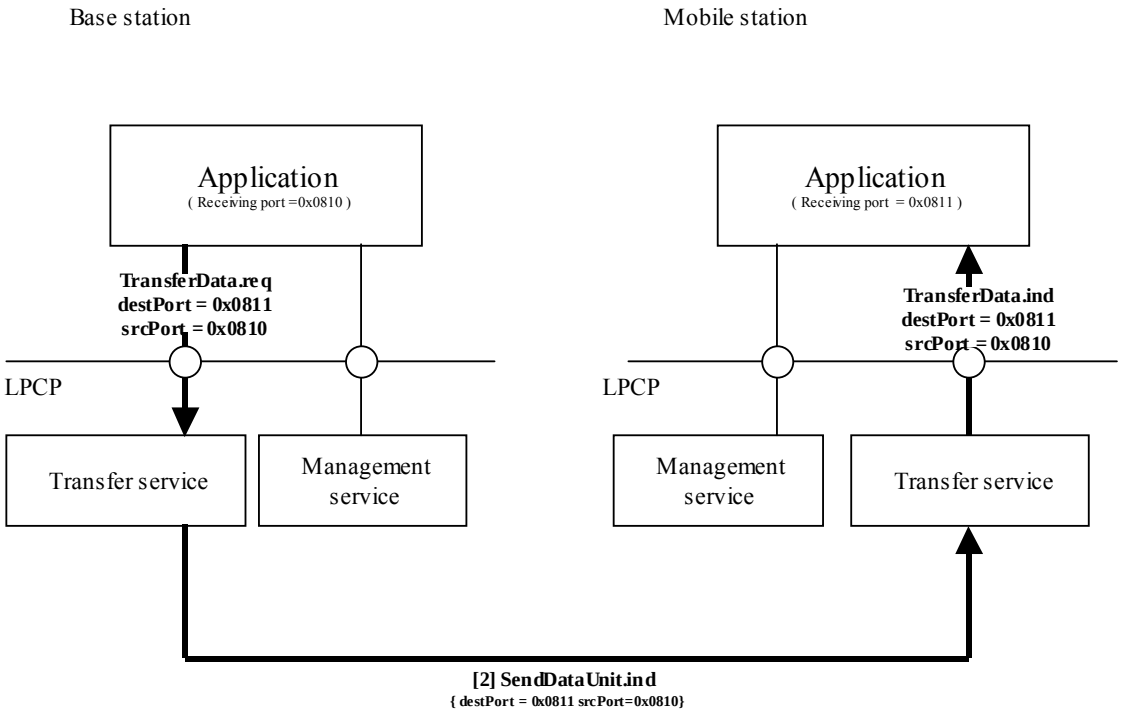


Figure H3-11 — Example of Data Transfer Operation (base station → mobile station)

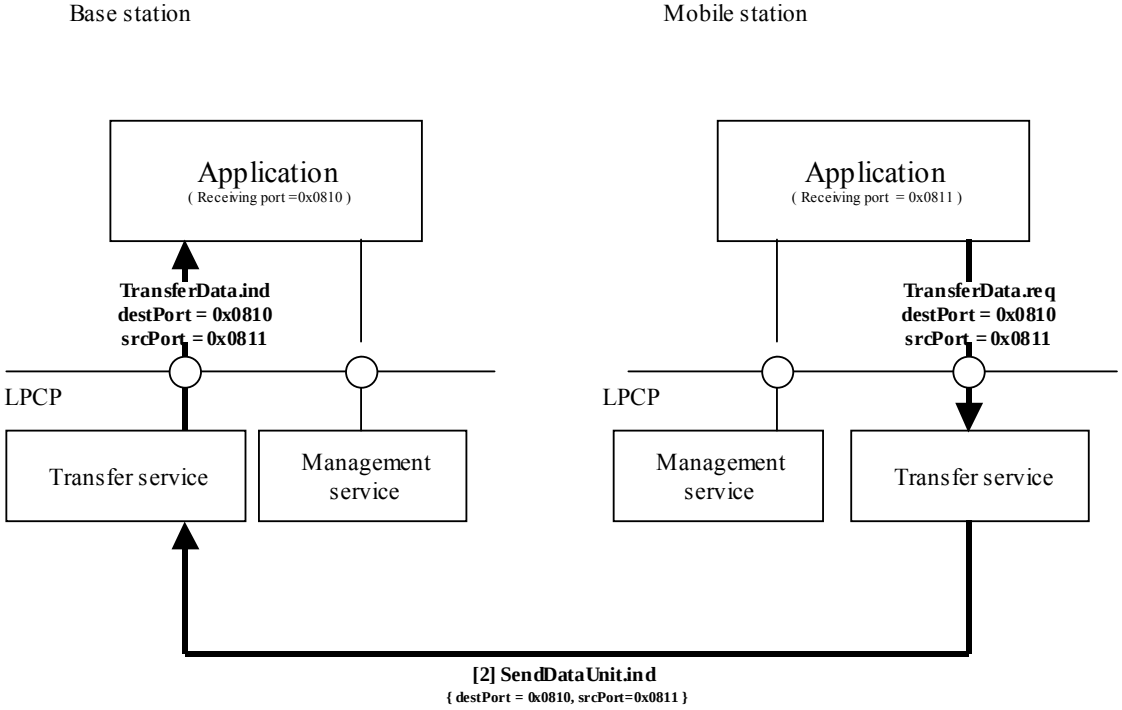


Figure H3-12 — Example of Data Transfer Operation (mobile station → base station)

H3.4.3 Broadcast Type Service Process Procedure

Figure H3-13 to Figure H3-15 show examples of process procedure for the broadcast type service.

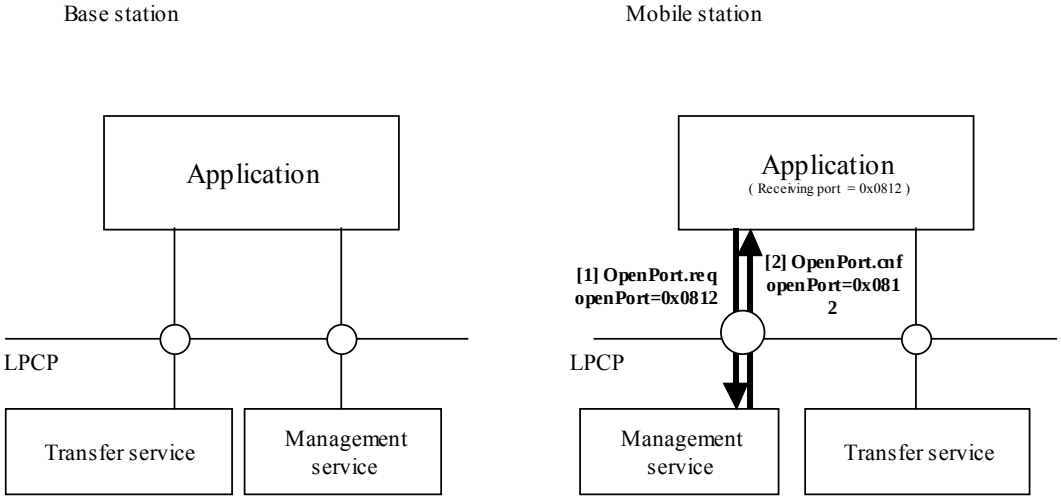


Figure H3-13 —Application Registration Operation

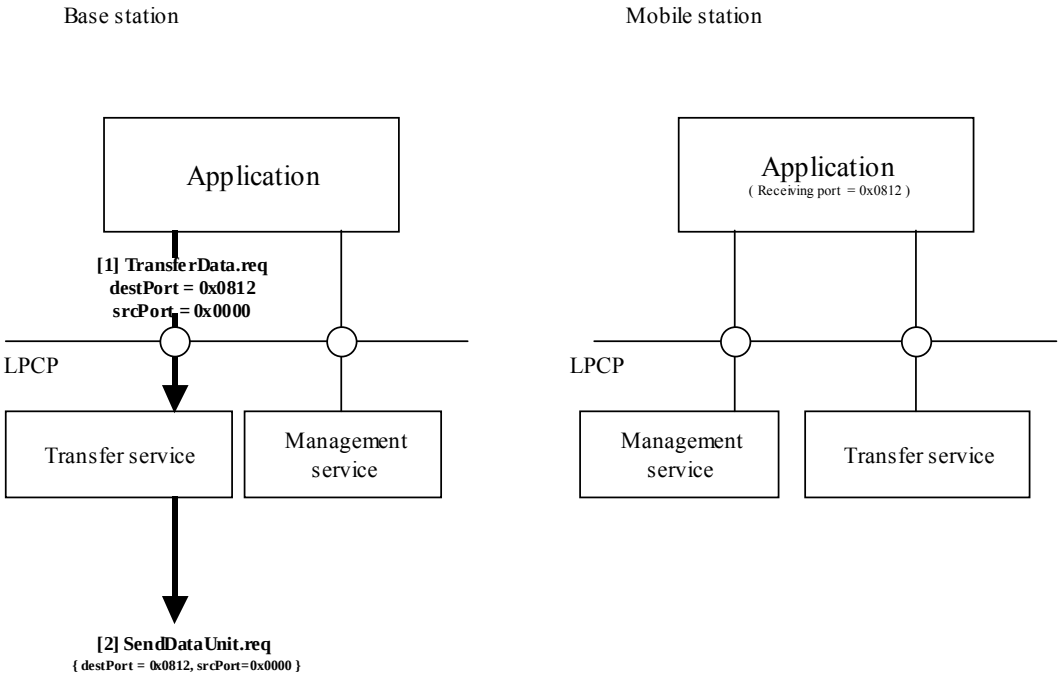


Figure H3-14 — Startup of Application on Base Station Side

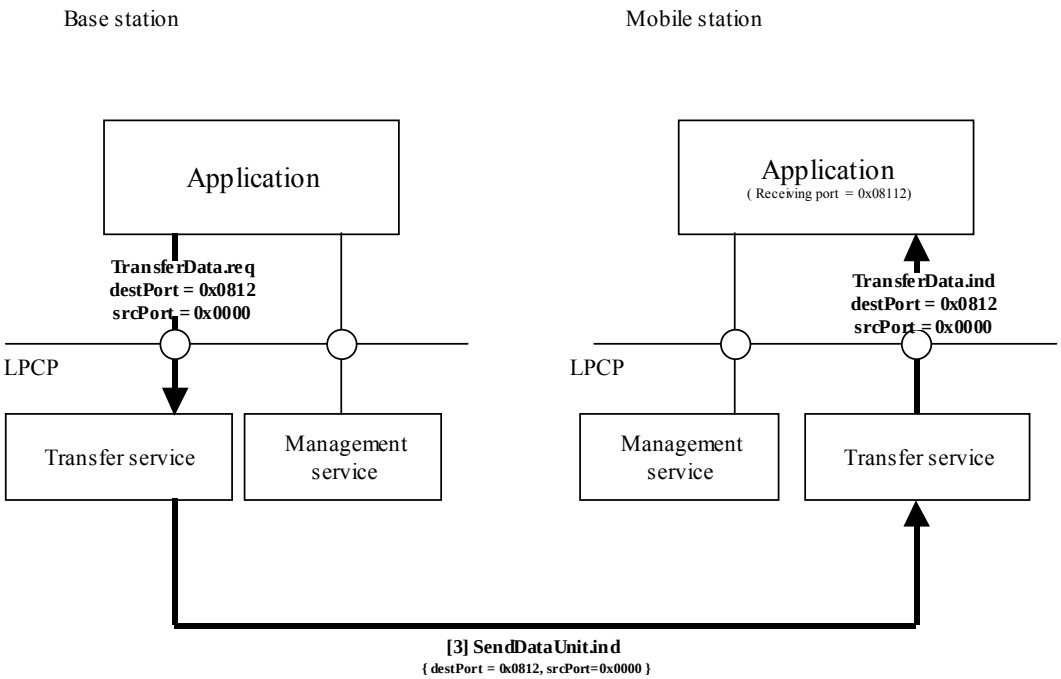


Figure H3-15 — Entry into Communication Process

Annex I
(informative)
Local Port Protocol (LPP)

I1 Function of LPP

The local port protocol (LPP) is a transaction-oriented protocol. It is located between the LPCP and non-network type applications, extends the functions of the LPCP, and provides the transaction service and connection management services described below to non-network type applications in the DSRC mobile stations/base station to improve the efficiency of the application building as shown in Figure I1-1. The LPP consists of the transaction service entity, which extends the communication functions of the LPCP, and the connection management service entity, which manages the communication status such as initial set up and disconnection. Each service has the following functions:

(1) Transaction service entity
   (a) Data exchange functions per transaction
   (b) Unidirectional data-sending transaction service
   (c) Request-response type transaction service
   (d) Data resend function (option)
   (e) Message segmentation/re-assembly function (option)
   (f) Transaction abortion function

(2) Connection management service entity
   (a) Communication connection query service
   (b) Communication disconnection notice service
   (c) Accept port query service

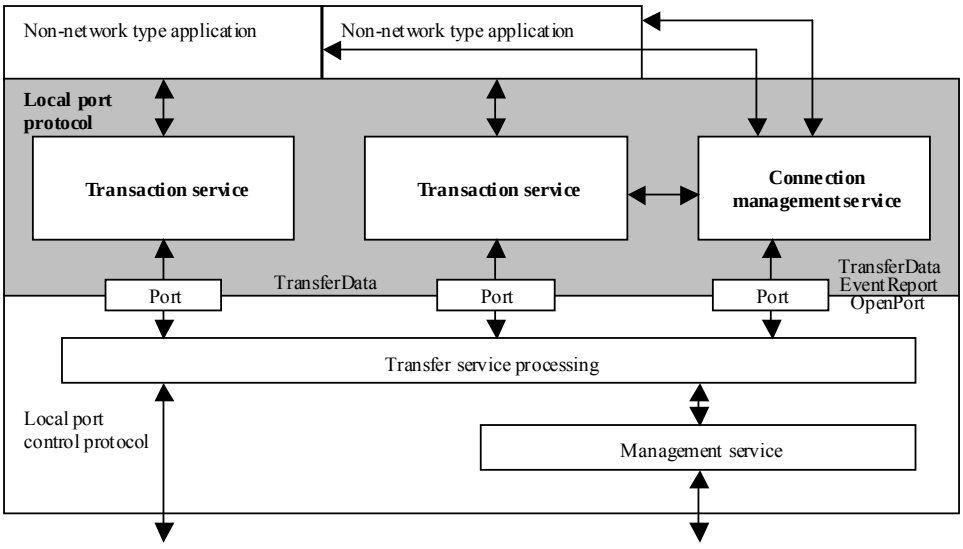


Figure I1-1 — Overview of Local Port Protocol (LPP)

I1.1 Transaction Services

I1.1.1 Data Exchange per Transaction

The LPP exchanges the application data for each single transaction.

Each transaction is identified by the transaction ID as shown in Figure I1-2. By this function, the system can handle situations in which two or more transactions exist simultaneously in the same application.

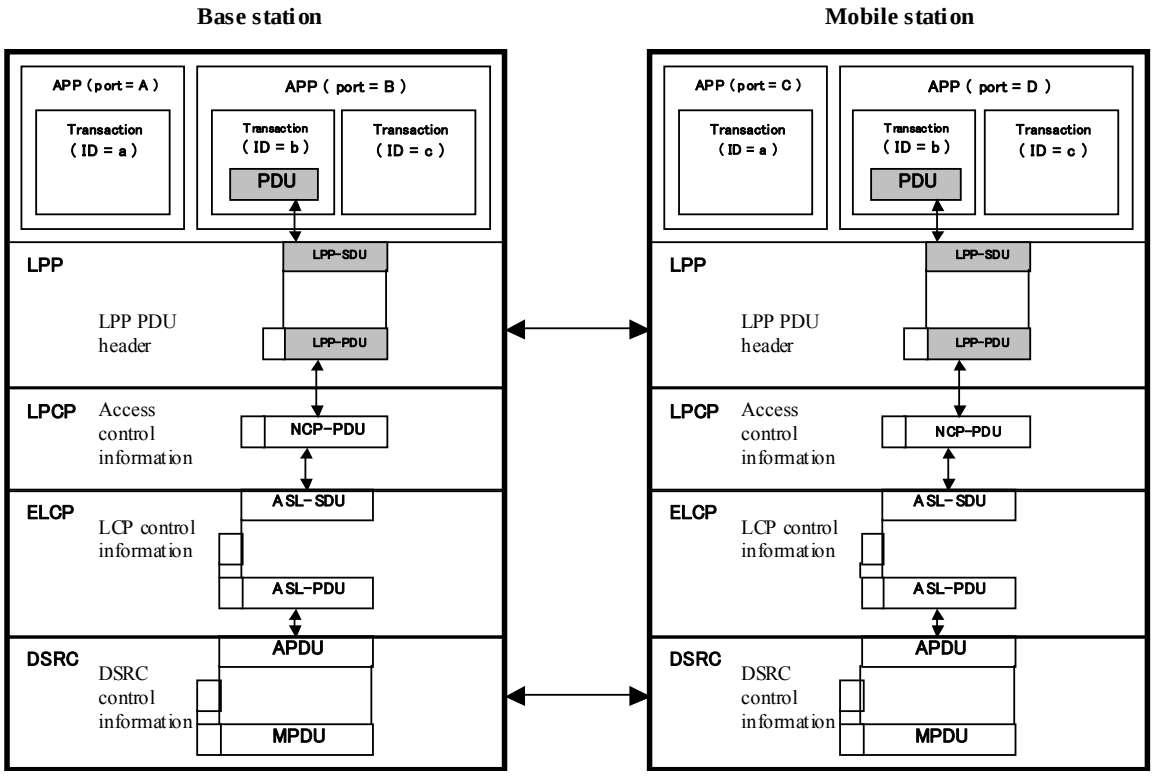


Figure I1-2 — Example of Data Exchange between Transactions in LPP

The transaction ID is assigned by the following method:

(1) The ID consists of 16 bits.
(2) The first bit indicates the transaction starting side. (“0” indicates a mobile station, and “1” indicates the base station.)
(3) Every time a new transaction is created, the ID is incremented by “1”.

I1.1.2 Two Types of Transaction Services

The LPP provides the following two types of transaction services:

(1) Unidirectional data-sending transaction service
(2) Request-response type transaction service

Each transaction service is selected according to the communication requirements of each application. Accordingly, the optimum communication service for each application is available.

(1) Unidirectional data-sending transaction service

The LPP provides data-sending service to non-network type applications on both the base and mobile stations as shown in Figure I1-3.

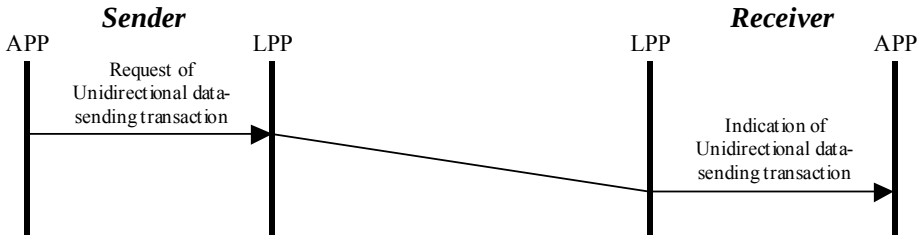


Figure I1-3 — Example of Unidirectional Data-sending Service

(2) Request-response type transaction service

The LPP notifies the counterpart of a message, and acquires the returned value for the message. This service can be used such as to call a method on a remote station. (Refer to Figure I1-4.)

This transaction is available only for point to point communication.

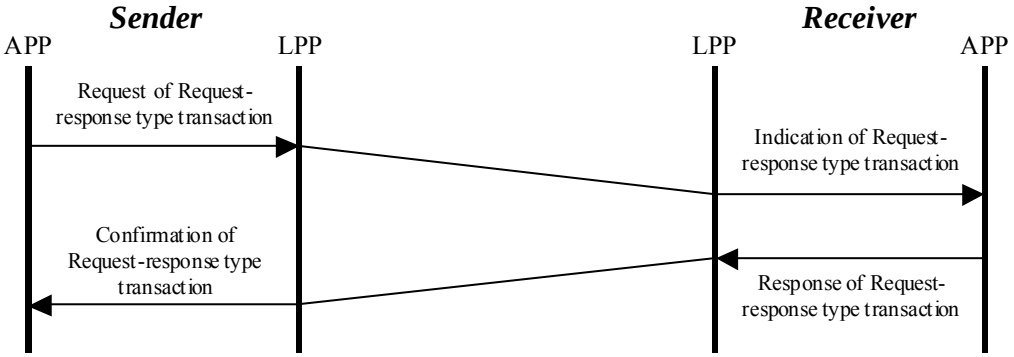


Figure I1-4 — Example of Request-response Type Transaction Service

I1.1.3 Data Resend Function (Optional)

This function is provided to ensure the reliability of communication, and controls resending of data using the resend timer and resend counter. When the resend timer expires, the LPP resends the data (up to the maximum number of times for resending) to ensure communication reliability as shown in Figure I1-5. This function can be applied to request and response data, and an application specifies whether or not to enable this function. The sequence is as follows:

When sending a packet, the LPP starts the resend timer and sets the resend counter to "0".
If data acknowledgement is not received before expiration of the resend timer, the LPP increments the resend counter and sends the packet again.

If the resend counter exceeds the maximum number of times for resending, the LPP concludes the transaction and notifies the application of transaction conclusion.

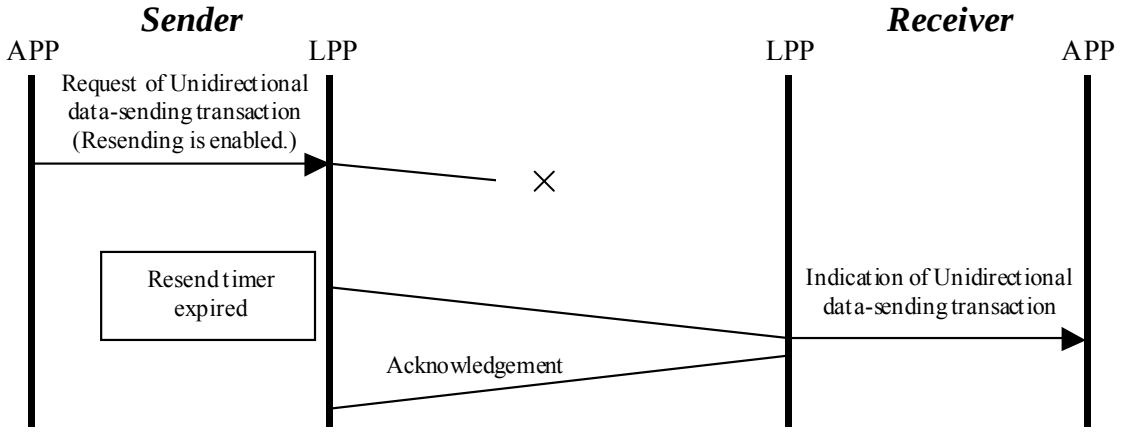


Figure I1-5 — Example of Data Resending

In a transaction using the data resend function, the PDU received previously may be received again due to acknowledgement not being received, etc. The LPP checks for such duplicate receiving using the transaction ID as shown in Figure I1-6. The duplicate receiving check method is implementation issue, and is not specified in this specification.

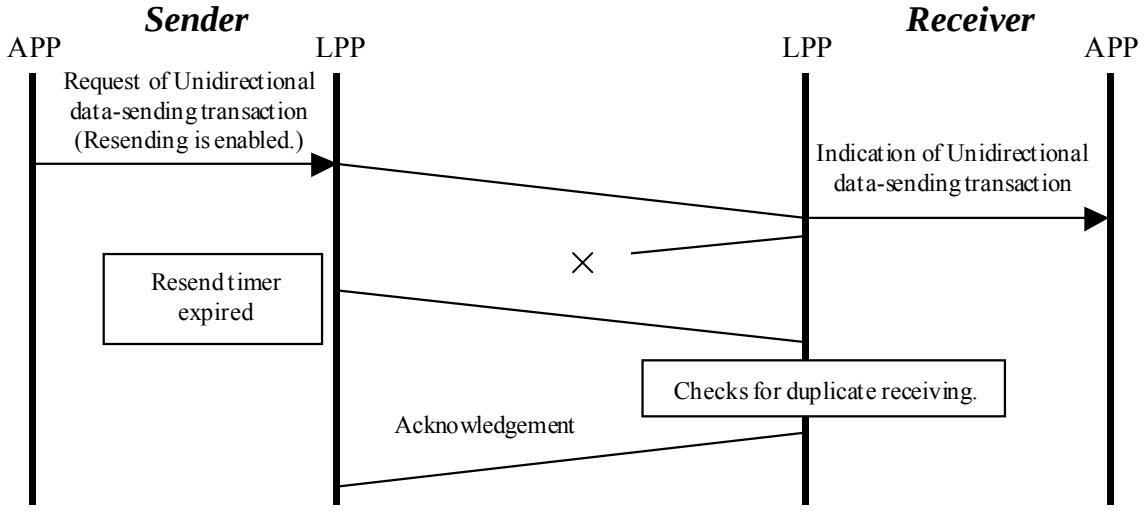


Figure I1-6 — Example of Duplicate Receiving Check

Although this resend function is optional, the response of acknowledgement (ACK) and duplicate receiving check function is mandatory. The resend function is available only for point to point communication.

I1.1.4 Segmentation/re-assembly of Message

This function segments and assembles a message so that the sending interface for a message, which exceeds the MTU of the LPCP can be provided to an application.

Figure I1-7 shows the message communication procedure using the segmentation/ re-assembly function. When the LPP receives a message which exceeds the MTU of the LPCP from an application, the LPP segments the LPP-SDU into segments (packets) of the size less than the MTU of the LPCP (SUL: segment unit for LPP, 508 octets), then transfers these packets in turn to the LPCP. In this process, segmented packets are added in the sending queue of the DSRC-ASL, and then transferred in turn to the DSRC layer 7. At this time, because it is postulated that the sending queue of the DSRC-ASL will overflow, the LPP sends again packets whose sending has failed and controls the flow to assure that all packets can be sent.

The receiving side acquires in turn the segmented packets passed from the LPCP, and adds them in the receiving queue prepared by the application on the receiver. At this time, it is not certain that each packet is stored in the receiving queue in the sending order due to the resend processing in layer 2 and the other. Accordingly, the receiver judges the assembly order based on the ordering number assigned to each packet, and then assembles each packet into the PDU. After receiving all packets, the receiver returns acknowledgement to the sender.

Some packets may be missing due to overflow in the receiving queue in the DSRC-ASL or data loss in the DSRC, so it is not certain that all sent packets reach the LPP in the receiver.

In this case, loss of one packet means loss of the entire message data. Therefore the following method ensures the arrival of the entire message. If a packet has not been received when the final packet is received, Receiver notifies those packets by negative acknowledgement, and the sender sends missing packets included in negative acknowledgement again (selective resend process). With regard to loss of the final packet, its arrival is ensured by the normal resend process. The same control is applied for packets sent by selective resending. Figure I1-8 shows an example of selective resend processing.

Because it is postulated that the size of the receiving queue required in each application is considerably different, each application prepares its own receiving queue for this function. Accordingly, with regard to transactions requiring disassembly and assembly, it is be possible to perform only one such transaction at a time for each sending destination (which is identified by the link address and destination port number).

In the case of sending data to a broadcast address, return of arrival acknowledgement, selective resend processing and final segment resend control are not performed. Instead, the required reliability of communication is assured by the transaction re-execution request.

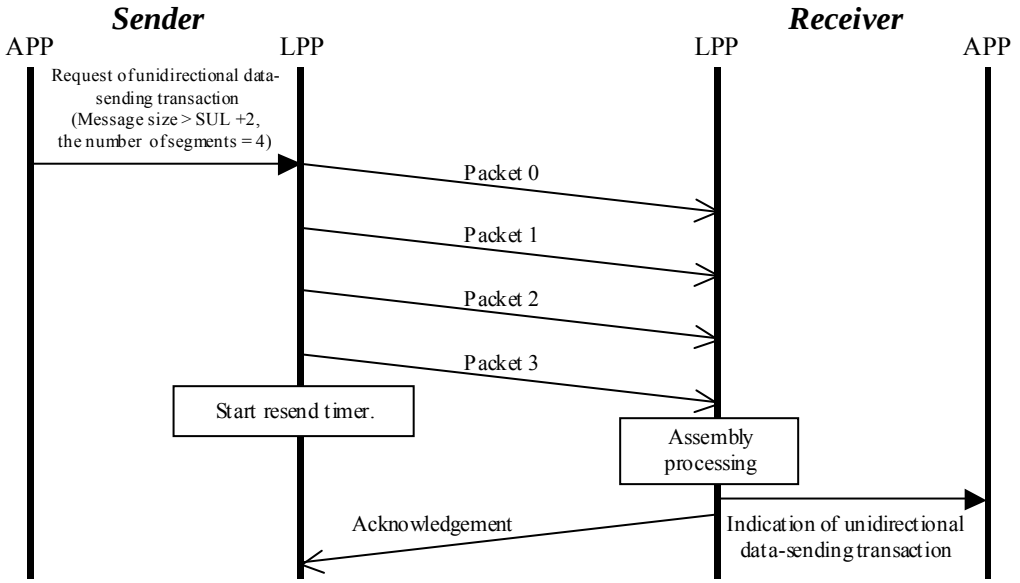


Figure I1-7 — Example of Segmentation/re-assembly of Message

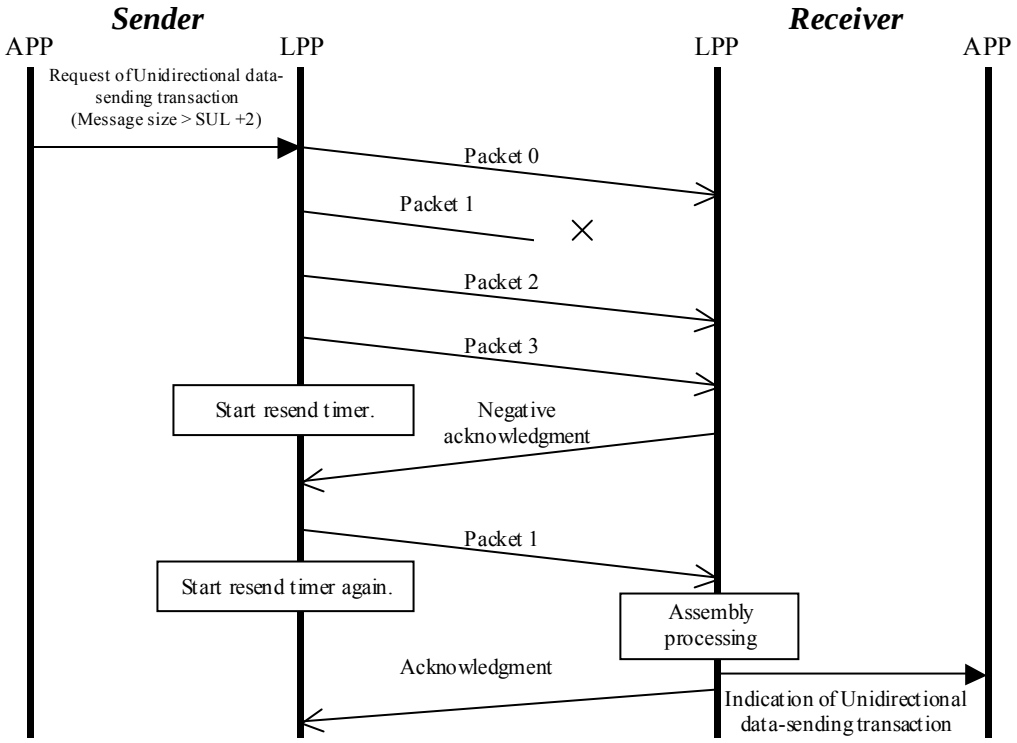


Figure I1-8 — Example of Selective Resend Process

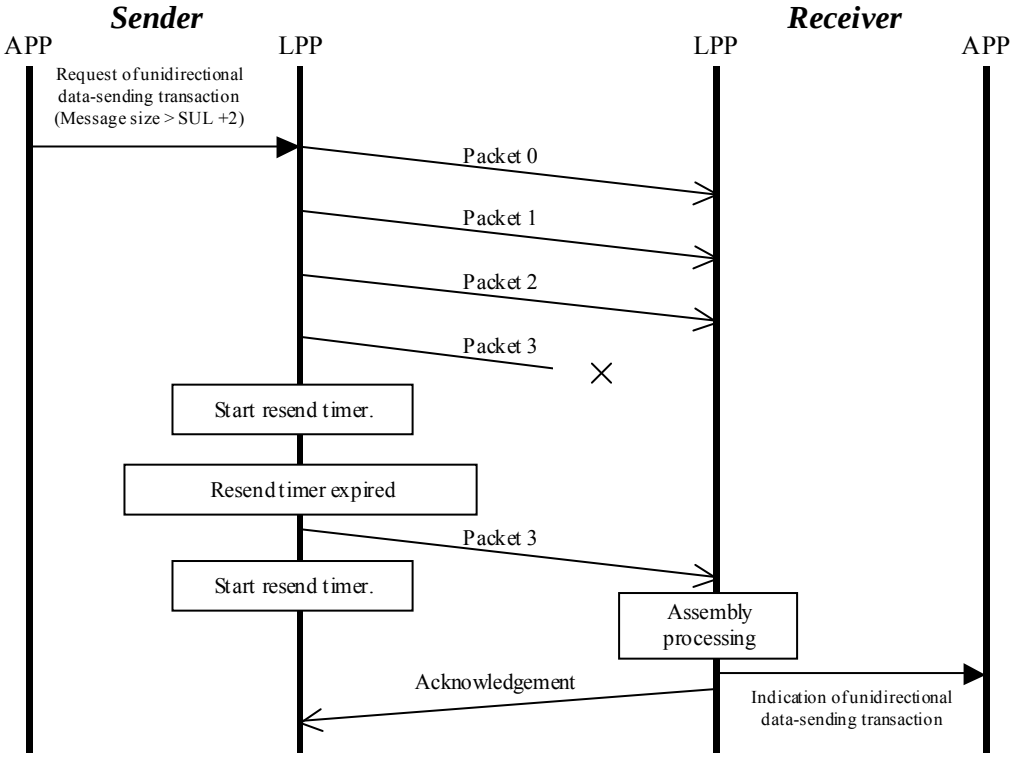


Figure I1-9 — Example of Last Packet Resend Process

I1.1.5 Transaction Abortion Function

Abortion of a transaction can be requested from an application or the system as shown in Figure I1-10. The LPP performs one of the following processes according to the transaction status at the time of request.

(1) When a message has not been sent, the LPP abandons the message.
(2) When a message has been already sent or is being sent, the LPP abandons all data related to the transaction, and notifies own station and remote station that the transaction has been aborted.
(3) When the transaction abortion request is received from the remote station, the LPP notifies the application that the transaction will be aborted, and then abandon all data related to the transaction.

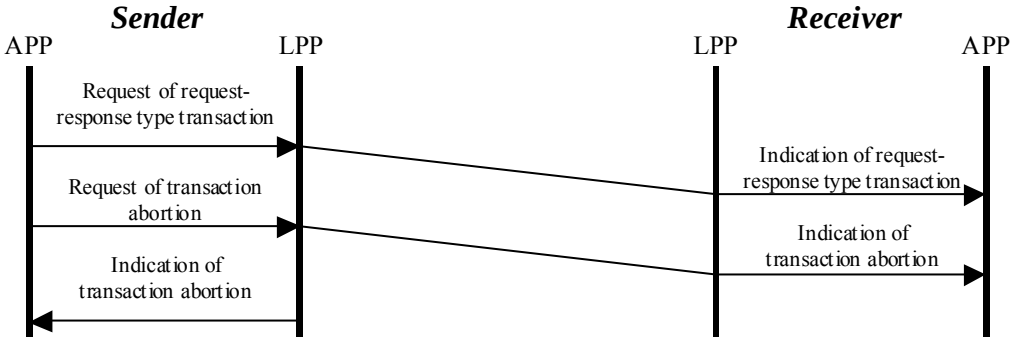


Figure I1-10 — Example of Transaction Abortion

And in the following cases:

(1) when the DSRC is disconnected or
(2) when the destination port is not an accept port,

The LPP does not start the transaction, but notifies the application that the request has failed so that unnecessary communication is omitted.

I1.2 Connection Management Service

In the connection management service, the LPP provides the start/end trigger of a communication to the application by the following services:

(1) The LPP manages and monitors the DSRC connection status, then reports the connection status and notifies new connections and disconnections by requests from an application.
(2) The service, which reports the situation of port by the request from an application or notifies to an application that the attribute of port is changed to acceptable. To provide this service, the LPP receives the accept port list event from the LPCP when the DSRC is connected and the LPP manages accept port numbers in the remote station by notifying accept port numbers and reject port numbers between the connection management

service on the base station and the mobile station when accept ports are changed.

The connection management service is regarded as similar to an application on the local port control protocol. For sending and receiving events between the connection management service on the base station and the mobile station, the data transfer service of the local port control protocol is used. The port number used by the connection management service is 0x0FFF.

I1.2.1 DSRC Connection Query Service

The LPP has a function to query whether or not DSRC is connected.

Two types of services, reference service and notification service, are specified. In the reference service, the LPP immediately returns the DSRC connection status at the time of query. In the notification service, the LPP waits for connection if the DSRC is not connected, then provides notification of connection at the time of connection.

I1.2.2 DSRC Disconnection Notify Service

This service notifies the disconnection of DSRC to an application, which requires notification.

I1.2.3 Accept Port Query Service

This service queries whether or not a port is present as receiving port in the remote station. There are three types of port statuses as follows:

(1) Accept port
This port is opened by the remote station as a data-receiving port.

(2) Reject port
This port is not opened by the remote station as a data-receiving port.

(3) Unknown port
It is not known whether or not this port is opened by the remote station as a data-receiving port. This is the initial status.

Two types of accept port query services, reference service and notification service, are specified. In the reference service, the LPP immediately returns the port status at the time of query. In the notification service, the LPP waits until the queried port changes to accept status, and then notifies the status when the accept port notification is received from the remote station (If it is already known that the queried port is a accept port, the service replies immediately.).

In order to enable the two types of services above, the management service in the LPP on the base station and a mobile station receives the accept port list event from the LPCP when the DSRC is connected, and then notify accept/reject port numbers in own station to remote station when accept ports are changed.

I2 Implementation Scope of LPP

Table I2-1 shows the implementation range of each function of the LPP on the bi-directional type service and broadcast type service specified in Annex H2.

Table I2-1 — Implementation Scope of LPP

		Broadcast type service		Bi-directional type service		
		Road traffic information*	Traffic / area guidance / sightseeing information	Collection of uplink information	Selective information provisioning	Charge / reservation
Functions of LPP	Unidirectional data-sending transaction service	M	M	M	M	M
	Request-response type transaction service	-	-	O	O	M
	Resend function	-	-	-	-	M
	Segmentation/re-assembly function	-	M	O	M	-
	Transaction abortion	-	-	-	O	M
	DSRC connection notification/Accept port query service	-	-	M	M	M
	DSRC disconnection notification service	-	-	M	M	M
	Receiving port registration	M (mobile)	M (mobile)	M	M	M
	Receiving port deregistration	O (mobile)	O (mobile)	O	O	O

*NOTE When the sending data is disassembled into segments whose size is the MTU or less.

Symbol Description

M	Mandatory
M (mobile/base)	Mandatory only in mobile/base station
O	Optional
O (mobile/base)	Optional only in mobile/base station
-	Not applicable

I3. Service Specification

I3.1 Notation

Table I3-1 shows the list of primitive types specified in this document.

Table I3-1 — Primitive Types

Primitive type	Abbreviation	Description
Request	req	Used when an upper layer requests a service from a lower layer.
Indication	ind	Used when a lower layer notifies an upper layer of a service from the counterpart.
Response	res	Used when an upper layer gives a response to a lower layer for a service to the counterpart.
Confirm	cnf	Used when a lower layer notifies an upper layer that the requested service is completed.

Table I3-2 shows the list of parameter types used in the primitive definition table in this document.

Table I3-2 — Parameter Types

Symbol	Description
M (mandatory)	Mandatory parameter
C (conditional)	Parameter specified when specified in the immediately preceding primitive (“req” in the case of “ind”, and “res” in the case of “cnf”)
O (optional)	Optional parameter
(=)	Indicates that the parameter value is equivalent to the value of the immediately preceding primitive (“req” in the case of “ind”, and “res” in the case of “cnf”).

I3.2 Service Primitives for Transaction Service

The LPP prepares the following two types of primitives as the transaction service for applications.

(1) Invoke (transaction start primitive)
(2) Abort (transaction abortion primitive)

I3.2.1 Invoke

(1) Function
The invocation of the “Invoke” service results in the generation of a new transaction.
Every transaction is started by calling this primitive.

(2) Semantics of Service Primitive

Table I3-3 — Parameters of Invoke

Primitive parameter	Invoke			
	req	ind	res	cnf
LinkAddress	M	M(=)	-	-
Source Port	M	M(=)	-	-
Destination Port	M	M(=)	-	-
UserDataSize	M	M(=)	M	M(=)
UserData	O	C(=)	O	C(=)
Transaction Type	M	M(=)	-	-
Require Ack	O	-	O	-
Result Timeout	O	-	-	-
Handle	M	M	M	M

The parameter “Link Address” is specific to the link address used in the DSRC.

The parameter “Source Port” is specific to the local port number of the sending source application.

The parameter “Destination Port” is specific to the local port number of the sending destination application.

The parameter “User Data Size” is specific to the size of the User Data (units: octets).
The “User Data” parameter is specific to the sending data.

The parameter “Transaction Type” is specific to the transaction service type. The value is shown in Table I3-4.

The ” parameter “Require Ack is specific to the flag indicating whether resend processing is enabled. “0” indicates that resend processing is not necessary, and “1” indicates that the resend processing is necessary.

The parameter “Result Timeout” is specific to the timeout time concern with the receiving of PDU “Result” or PDU “ResultSegment” in the request-response type transaction service. If PDU “Result” or PDU “ResultSegment” does not arrive within this time after “Invoke.req” is executed, the transaction is aborted.

The “Handle” parameter is specific to the ID assigned to identify the transaction in the local station. The “Handle” specified here is required to satisfy the following conditions:

(a) On the Sender, the transaction ID is determined uniquely from the “Handle” and “Source Port”.
(b) On the Receiver, “Link Address”, “Source Port” and transaction ID is determined uniquely respectively from “Handle”.

If the same “Handle” as that in the broadcast communication executed immediately preceding the current communication is specified in the broadcast communication, it is handled as a transaction re-execution request.

Table I3-4 — Transaction Types

Transaction Type	Description	Remarks
0	Data-sending transaction service	
1	Request-response type transaction service	

I3.2.2 Abort

(1) Function
The invocation of “Abort” service results in the abortion of an active transaction.

(2) Semantics of Service Primitive

Table I3-5 — Parameters of Abort

Primitive parameter	Abort	
	req	ind
Abort Type	O	C(=)
Abort Code	O	C(=)
Handle	M	M

The parameter “Abort Type” is specific to the type for aborting: system error (0) or user request (1).

The parameter “Abort Code” is specific to the reason for aborting the transaction. The details of system errors are shown in Table I3-6.

The parameter “Handle” is the ID assigned to identify the transaction in the local station.

Table I3-6 — Abort Code List

Abort Code	Code	Description
Unknown	0x00	-
Protocol error	0x01	The received PDU structure is abnormal.
TID is invalid.	0x02	The TID is invalid.
Transaction service is not supported.	0x03	The Receiver does not support the request-response type transaction service.
LPP version is different.	0x04	The LPP version is different between the Sender and the Receiver.
Receive buffer overflow	0x05	The receive buffer has overflowed.
MTU error	0x06	Because the send data exceeded the MTU in the LPCP, the transaction could not be started (when segmentation/re-assembly processing is not supported).
Resend timer timeout	0x07	The resend timer expired, and the transaction was aborted.
Result timer timeout	0x08	The result timer expired, and the transaction was aborted.
Link Address error	0x09	(Point to point) The vehicle is not present inside the zone. (Broadcast) The broadcast address is illegal.
Destination port error	0x0A	The destination port number is not present in the counterpart.
LPP is not supported.	0x0B	The DSRC-ASL does not support this protocol.
Aborted by DSRC-ASL.	0x0C	Because there was no space in the send queue in the DSRC-ASL, the requested service was aborted.
Transaction was not started.	0x0D	Because the number of transactions exceeded the number which can be executed at the same time, the transaction could not be started.
Under segmentation/assembly processing	0x0E	A transaction used segmentation/re-assembly processing is being executed.
Reserved for future use	0x0F to 0xFF	Reserved for future use

I3.3 Connection Management Service Primitive

As the connection management service, the LPP provides the following four types of primitives for applications.

(1) Connect (transaction start query/notification primitive)
(2) Disconnect (DSRC disconnection notification primitive)
(3) RegisterPort (Port registration primitive)
(4) DeregisterPort (Port deregistration primitive)

I3.3.1 Connect

(1) Function
The invocation of “Connect” service results in the query whether or not a transaction may be

started. The “Connect.cnf” primitive notifies the result of query by “Connect.req”. The results are the DSRC connection status, link address and accept port number in the remote station, which is indicated by the link address.

(2) Semantics of Service Primitive

Table I3-7 — Connect Parameters

Primitive parameter	Connect			
	req	ind	res	cnf
Querist Port	M			-
Query LID	O			-
Query Port	O			-
TimeOut	O			-
Connected LID	-			M
Accept Port	-			M

The parameter “QueristPort” is specific to the local port number of the querist application.

The parameter “Query LID” is specific to the link address to be queried. When this parameter is specified, this primitive results in the query for a connected link. When this parameter is not specified, this primitive results in the waiting for new connection. When both “Query LID” and “Query Port” are omitted, “Connect.cnf” is issued immediately after the DSRC connects (fast connection mode). When “Query Port” is specified, “Connect.cnf” is issued after the accept port notification is received (normal connection mode).

The parameter “Query Port” is specific to the destination local port number to be queried.

The parameter “TimeOut” is specific to the timer value of “Connect” service. When the DSRC is connected during this timer is operating, “Connect.cnf” is issued immediately. If not this parameter, the timer value is “∞”.

The parameter “Connected LID” is the same link address as “Query LID” when “Query LID” is specified and its link address is connected. When “Query LID” is specified and its link address is not connected or when “Query LID” is not specified and a new connection is not achieved within the time specified by “TimeOut”, “Connected LID” indicates “-1”.

The parameter “Accept Port” is the accept port number held by the remote station indicated by “Connected LID”. When the port number is specified by “Query Port”, “Accept Port” indicates only the specified port number. When the specified local port number is a reject port number, “Accept Port” indicates “-1”. When “Query Port” is omitted, “Accept Port” indicates “0”.

I3.3.2 Disconnect

(1) Function
The primitive “Disconnect” is used to notify an application that the DSRC is disconnected.

(2) Semantics of Service Primitive

Table I3-8 — Disconnect Parameters

Primitive parameter	Disconnect	
	req	ind
Link Address	-	M

The “Link Address” parameter is the link address used in the DSRC.

I3.3.3 RegisterPort

(1) Function
The primitive “RegisterPort” is used to register a receiving port for the LPP.

(2) Semantics of Service Primitive

Table I3-9 — RegisterPort Parameters

Primitive parameter	RegisterPort
	req
PortNo	M
BulkArea	O
BulkAreaSize	O

The parameter “PortNo” is a receiving local port number.

The parameter “BulkArea” is the area to assemble disassembled segments of a message. This parameter is optional.

The parameter “BulkAreaSize” is the size of “BulkArea”. This parameter is optional.

I3.3.4 DeregisterPort

(1) Function
The primitive “DeregisterPort” is used to deregister a receiving port for the LPP.

(2) Semantics of Service Primitive

Table I3-10 — DeregisterPort Parameters

Primitive parameter	DeregisterPort
	req
PortNo	M

The parameter "PortNo" is a receiving local port number to be deregistered.

I4 Protocol Data Unit (PDU)

I4.1 PDUs in Transaction Service

PDUs used in the transaction service are classified into 7 types as shown in Table I4-1. The PDU used in the transaction service consists of the header area defined for each PDU type and the data area, which indicates the application data. Figure I4-1 shows the basic structure of the LPP-PDU.

Table I4-1 — PDU Type List

PDU type	Usage scenario
Invoke	Used in the primitive "Invoke.req".
Result	Used in the primitive "Invoke.res".
Acknowledgement	Used in arrival acknowledgement.
Abort	Used when a transaction is aborted (due to the primitive "Abort" or system error).
InvokeSegment	Used when the message size exceeds the MTU in the LPCP in the primitive "Invoke.req".
ResultSegment	Used when the message size exceeds the MTU in the LPCP in the primitive "Invoke.res".
Nack	Used in selective resend processing for segmentation/assembly processing.

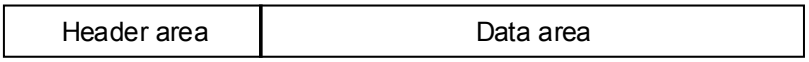


Figure I4-1 — Basic Structure of LPP-PDU

I4.1.1 Invoke PDU

(1) PDU Type
This field indicates the PDU type. In the Invoke PDU, this field always indicates "Invoke(1)".

(2) Version
This field indicates the LPP version. The current version is 0x00.

(3) TT
The TT indicates the LPP Transaction Type.

(4) RA
The RA indicates weather or not resend processing is enabled. (1: enable, 0:disable)

(5) RD
The RD indicates whether or not the data is resent data. (1:resent 0: not resent.)

(6) TID
This field indicates the transaction ID.

Table I4-2 — Invoke Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = Invoke (0x01)			Version		TT	RA	RD
2	TID							
3								

I4.1.2 Result PDU

(1) PDU Type
This field indicates the PDU type. In the Result PDU, this field always indicates "Result(2)".

(2) RA
This field indicates for Require Ack, which is the flag indicating whether or not resend processing is enabled. (1:resent 0: not resent.)

(3) RD
This field indicates for Retransmission Data, which is the flag indicating whether or not the PDU are resent PDU.

(4) TID
This field indicates the transaction ID.

(5) RES

This field is reserved.

Table I4-3 — Result PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = Result (0x02)			RES	RES	RES	RA	RD
2	TID							
3								

I4.1.3 Acknowledgement PDU

(1) PDU Type
This field indicates the PDU type. In the PDU Acknowledgement, this field always indicates "Ack(3)".

(2) RD
This field indicates for Retransmitted Data, which is the flag indicating whether or not the data are resent data. (1:resent 0: not resent.)

(3) TID
This field indicates the transaction ID.

(4) RES
This field is reserved.

Table I 4-4 — Acknowledgement PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = Ack (0x03)			RES	RES	RES	RES	RD
2	TID							
3								

I4.1.4 Abort PDU

(1) PDU Type
This field indicates the PDU type. In the PDU "Abort", this field always indicates "Abort(4)".

(2) AT
This field indicates for Abort Type which is the flag indicating the reason for aborting.
("1":request from the user "0": system error.)

(3) TID
This field indicates the transaction ID.

(4) Abort Code
This field indicates the reason for aborting the transaction Abort code for system error is shown in Table I3-6.

(5) RES
This field is reserved.

NOTE The destination port number and sending source port number should be judged based on the TID.

Table I4-5 — Abort PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = Abort (0x04)			RES	RES	RES	RES	AT
2	TID							
3								
4	Abort Code							

I4.1.5 InvokeSegment PDU (optional)

(1) PDU Type
This field indicates the PDU type. In the InvokeSegment PDU, this field always indicates InvokeSegment(5).

(2) Version
This field indicates the LPP version. The current version is 0x00.

(3) TT
TT indicates for Transaction Type which is the flag indicating the transaction type.
("1": in the case of request-response type transaction service "0": in the case of data-sending type transaction service).

(4) FIN
This field indicates whether or not the segment is final.
("1": when the segment is final "0": in any other case)

(5) RD
RD indicates for Retransmitted Data, which is the flag indicating whether or not the data are resent data.
("1": resent "0": not resent)

(6) TID
This field indicates the transaction ID.

(7) Segment No
This field indicates the sequence number of the PDU.

Table I4-6 — InvokeSegment PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = InvokeSegment (0x05)			Version		TT	FIN	RD
2	TID							
3								
4	Segment No							
5								

I4.1.6 ResultSegment PDU (optional)

(1) PDU Type
This field indicates the PDU type. In the ResultSegment PDU, this field always indicates ResultSegment (6).

(2) FIN
This field indicates whether or not the segment is final.
("1": when the segment is final "0": in any other case)

(3) RD
RD indicates for Retransmitted Data, which is the flag indicating whether or not the data are resent data.
("1" : resent "0" : not resent)

(4) TID
This field indicates the transaction ID.

(5) RES
This field is reserved.

(6) Segment No
This field indicates the serial number of the PDU.

Table I4-7 — ResultSegment PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = ResultSegment (0x06)			RES	RES	RES	FIN	RD
2	TID							
3								
4	Segment No							
5								

I4.1.7 Nack PDU (optional)

(1) PDU Type

This field indicates the PDU type. In the PDU “Nack”, this field always indicates “Nack (7)”.

(2) RD
RD indicates for Retransmitted Data, which is the flag indicating whether or not the data are resent data.
(“1”: resent, “0”: not resent)

(3) TID
This field indicates the transaction ID.

(4) RES
This field is reserved.

(5) NumSeg
This field indicates the count of sequence numbers of PDUs not received yet.

(6) Segment Number List
This field indicates the list of sequence numbers of PDUs not received yet.

Table I4-8 — Nack PDU Header Information

Bit/Octet	7	6	5	4	3	2	1	0
1	PDU Type = Nack (0x07)			RES	RES	RES	RES	RD
2	TID							
3								
4	NumSeg = *n*							
5								
6 : 6+n*2-1	Segment Number List							

I4.2 PDUs in Connection Management Service

The connection management service of the LPP notifies the accept port list and reject port list to the remote station by using the transfer service of the LPCP when the DSRC is newly connected or when the number of accept ports has increased or decreased. PDUs used in these notifications are shown below and they are stored in the user data area in the LPCP PDU.

I4.2.1 Accept Port PDU

(1) Status
This field indicates the event type. This field always indicates “acceptPort (1)” in case of accept port list notification.

(2) AcceptPort
This field indicates accept port numbers.

Table I4-9 — Accept Port PDU

Bit/Octet	7	6	5	4	3	2	1	0
1	Status = acceptPort (1)							
2	AcceptPort							
3								

I4.2.2 Reject Port PDU

(1) Status
This field indicates the event type. This field always indicates rejectPort (2) in case of reject port list notification.

(2) RejectPort
This field indicates reject port numbers.

Table I4-10 — Reject Port PDU

Bit/Octet	7	6	5	4	3	2	1	0
1	Status = rejectPort (2)							
2	RejectPort							
3								

I5 Procedure

I5.1 Initial Set up Procedure

This paragraph describes the initial set up procedure for normal connection applications and fast connection applications using point to point communication. An application with broadcast communication can be executed without the initial set up procedure.

(1) Initial Set up Procedure for Normal Connection Applications
(a) Each application in a mobile station and the base station registers accept port numbers in the LPP by the port registration primitive (RegisterPort).

(b) The LPP updates the connection management table, and registers the accept port numbers specified in (a) and the connection management service port as the data receiving ports and the event receiving ports in the LPCP.

(c) Each application with “QueryLID” and “Query Port”, issues the “Connect.req” primitive, and waits for connection of the DSRC (blocking call).

(d) The connection management service of the LPP receives the event “Connection notice (96)” from the LPCP through the “EventReport.indication”.

(e) The connection management service of the LPP prepares the connection management table for the link address received through the primitive above. For an application

requiring fast connection, the LPP accepts thereafter transaction start requests to all ports at this link address and broadcast addresses until the accept port list of the remote station is received from the LPCP.

(f) When the connection management service of the LPP receives the accept port list from the LPCP through the "EventReport.indication", the LPP registers the accept ports in the connection management table at the link address specified by the primitive above. Thereafter, the LPP accepts transaction start requests to this link address only for these accept ports.

(g) The LPP notifies an application which issued the "Connect.req" primitive to a port number included in the accept port list received in (f) the link address and sending accept port numbers by the "Connect.cnf" primitive.

(h) When the application issues "Invoke.req" primitive with the link address specified by the "Connect.cnf" primitive or broadcast addresses and the destination port number, the transaction is started.

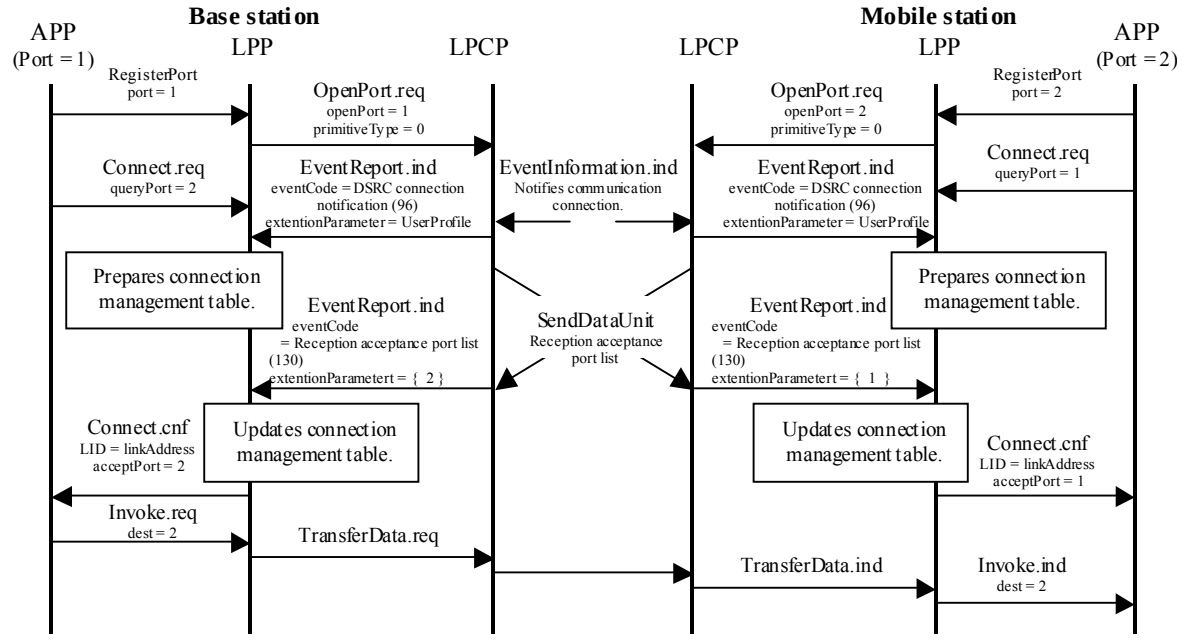


Figure I5-1 — Example of Initial Set up Procedure of LPP

(2) Initial Set up Procedure for Fast Connection Applications

(a) Each application in a mobile station and the base station registers accept port numbers to the LPP by the port registration primitive (RegisterPort).

(b) The LPP updates the connection management table, and registers the accept port

numbers in the LPCP.

(c) Each application issues the “Connect.req” primitive, without “QueryLID” and “Query Port”, and waits for connection of the DSRC.

(d) The LPP receives the event “connection notice (96)” from the LPCP through the “EventReport.indication”.

(e) The LPP prepares the connection management table for the link address received through the primitive above. For an application requiring fast connections, the LPP accepts thereafter transaction start requests for all ports for this link address and broadcast addresses until the accept port list of the remote station is received from the LPCP.

(f) The LPP notifies an application issuing the “Connect.req” primitive of (the link address) through the “Connect.cnf” primitive.

(g) Each application issues the “Invoke.req” primitive with the link address notified through the “Connect.cnf” primitive or broadcast addresses to start the transaction.

(h) If the port number specified in (g) is present in the remote station, this transaction succeeds. If the port number specified in (g) is not present in the remote station, the LPCP in the remote station notifies the event “The destination local port is invalid. (129)” through the event notification primitive, and the LPP updates the connection management table for this link address. In the case of Transaction Type = “1”, the LPP notifies the corresponding application that the transaction has failed through the “Abort.ind” primitive. If an “Invoke.req” primitive is given with this combination of link address and port number after that, the LPP notifies that the transaction is aborted through the “Abort.ind” primitive.

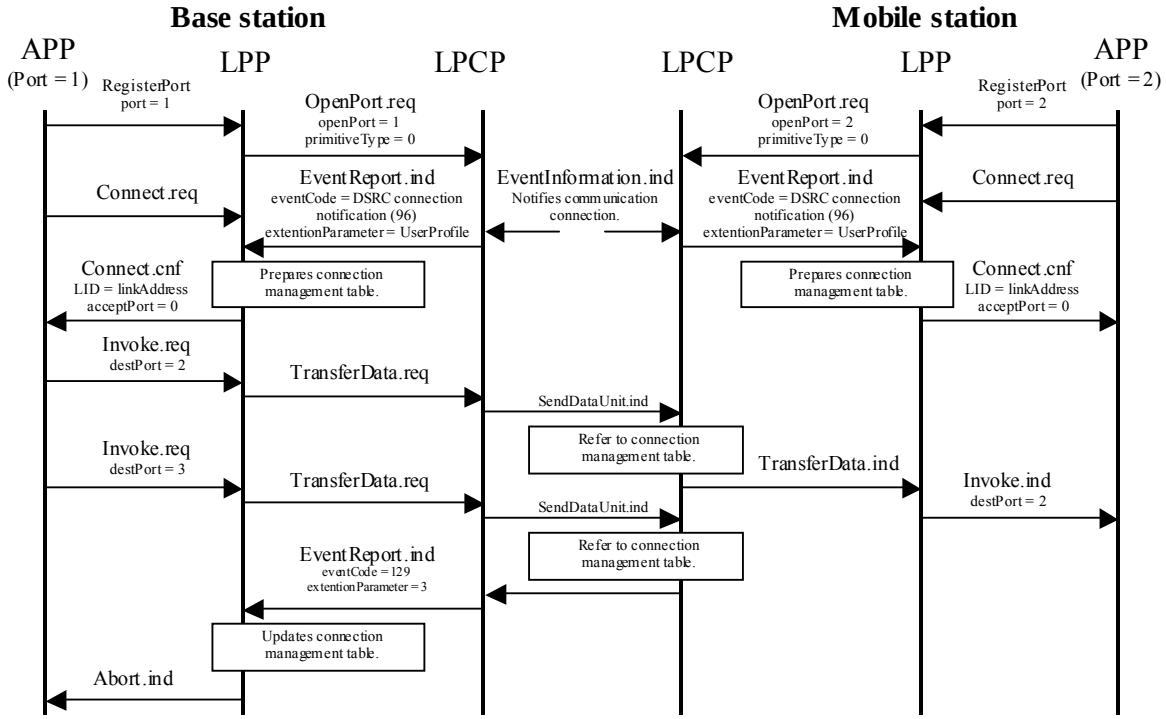


Figure I5-2 Example of Initial Set up Sequence in a Fast Connection Application

I5.2 Unidirectional Data-sending Transaction Service

(1) Send Processing

(a) When an application issues the “Invoke.req” primitive with Transaction Type = “0”, a data-sending transaction service is started.

(b) When the combination of specified link address and destination port number indicates a reject port, the LPP notifies the application of “destination port error” through “Abort.ind”, and the transaction is ended.

(c) When the specified message exceeds the MTU in the LPCP and segmentation/assembly processing is not supported, the LPP notifies the application of “MTU error” through the “Abort.ind” primitive, and the transaction is ended. The processing to be performed when segmentation/assembly processing is supported is described in I5.5.

(d) In any case other than (b) and (c), the LPP prepares the Invoke PDU with TT set to 0, and then sends it to the remote station through the “TransferData.request” in the LPCP. The processing to be performed when resend processing is valid is described in I5.4.

(2) Receive processing

(a) When the Invoke PDU sent in (1)-(d) through the “TransferData.indication” in the

LPCP is received, the LPP notifies the application of the received data through the "Invoke.ind" primitive.

Figure I5-3 shows an example of the sequence of the data transfer procedure in a data-sending transaction service.

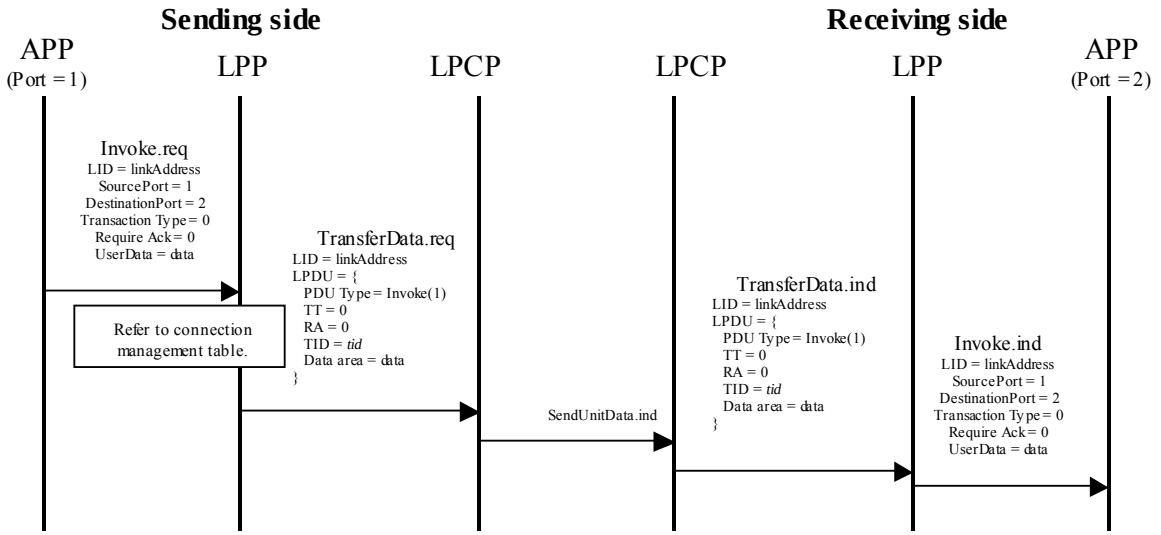


Figure I5-3 — Example of Unidirectional Data-sending Transaction Service

I5.3 Request-response Type Transaction Service

(1) Send Processing

(a) When an application issues the "Invoke.req" primitive with Transaction Type = 1, a request-response type transaction service is started.

(b) When the combination of specified link address and destination port number indicates a reject port, the LPP notifies the application of "destination port error" through "Abort.ind", and the transaction is ended.

(c) When the number of transactions exceeded executable at a time, the LPP notifies the application of "Transaction could not be started." through "Abort.ind", and the transaction is ended.

(d) In any case other than (b) and (c), the LPP prepares the Invoke PDU with TT set to 1, sends it to the remote station through the "TransferData.request" in the LPCP, starts up the Result timer (with the timeout value specified by "Invoke.req"), and waits to receive the Result PDU from the remote station.

(e) When the Result timer started up in (d) expires, the LPP prepares the Abort PDU with AT set to 0 and Abort Code set to 0x08, notifies the remote station of "Result timer timeout", and then notifies the application that the transaction has failed through the

Abort.ind primitive.

(f) When the Result PDU sent by the remote station through the “TransferData.indication” in the LPCP is received before the Result timer expires, the LPP stops the Result timer started up in (d), and then notifies the application of the response data through the “Invoke.cnf” primitive.

(2) Receive processing

(a) When the Invoke PDU sent by the remote station through the “TransferData.indication” in the LPCP is received, the LPP notifies the application of the receiving data through the “Invoke.ind” primitive, and waits to receive the “Invoke.res” primitive from the application.

(b) When the Abort PDU sent by the remote station through the “TransferData.indication” in the LPCP is received, the LPP issues the “Abort.ind” primitive, and notifies the application that the transaction has failed. The transaction is then ended.

(c) The application issues the “Invoke.res” primitive, and requests the LPP to send a response.

(d) The LPP prepares the Result PDU, and sends it to the remote station through the “TransferData.request” in the LPCP.

NOTE When segmentation/assembly processing is used in “Invoke.req” and “Invoke.res”, the send processing and receive processing is performed according to the procedure described in I5.5.

Figure I5-4 shows an example of the basic sequence in a request-response type transaction service. Figure I5-5 shows an example of the sequence when the Result timer expires.

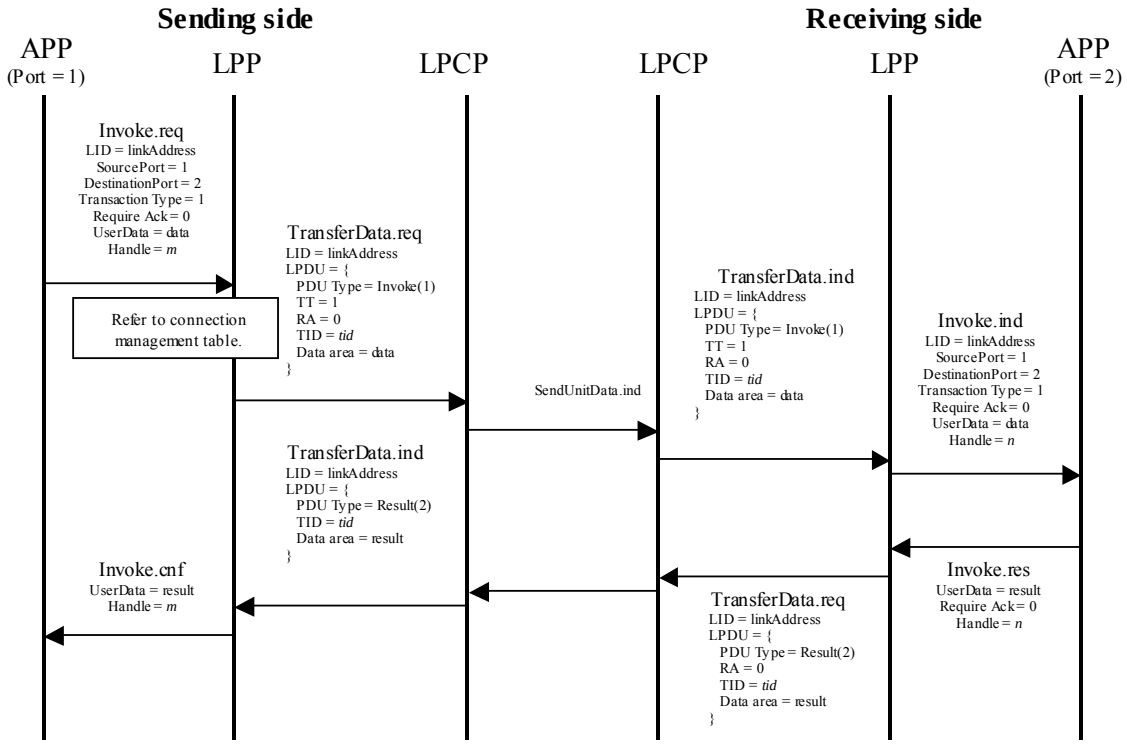


Figure I5-4 — Example of Request-response Type Transaction Service

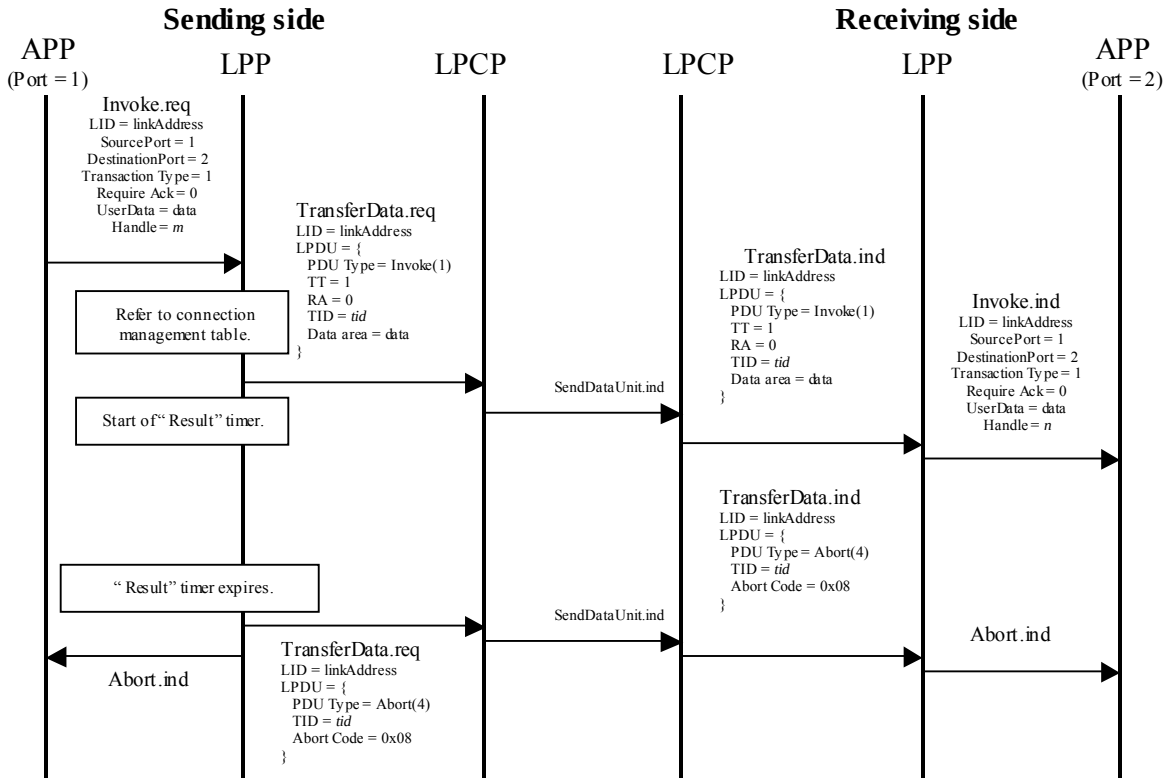


Figure I5-5 — Example of Request-response Type Transaction Service (Result timer expires)

I5.4 Resend Process (Optional)

Resend processing is used when “Require Ack” is set to “1” in “Invoke.req” and “Invoke.res”. This paragraph describes the sequence when resend processing is used in “Invoke.req” in a data-sending transaction. In the request-response type transaction service, the same processing is also available in “Invoke.res”.

(1) Send processing

(a) When an application issues the “Invoke.req” primitive with Require Ack = “1”, data transfer service in which resend processing is enabled is started.

(b) The LPP prepares the Invoke PDU with RA set to 1, sends it to the remote station using the “TransferData.request” in the LPCP, starts the resend timer, and waits to receive the PDU Acknowledgement from the remote station.

(c) When the resend timer started in (b) expires before the LPP receives the PDU Acknowledgement for some reason (such that the Invoke PDU sent in (b) does not arrive), the LPP sets the RD flag to 1 in the Invoke PDU sent in (b), sends the Invoke PDU again to the remote station, restarts the resend timer, and increments the resend counter.

(d) When the resend counter exceeds the maximum number of resend times after resend in(c), the LPP prepares the Abort PDU with AT set to “0” and Abort Code set to “0x07”, notifies the remote station of “Resend timer timeout”, notifies the application that the transaction has failed through the “Abort.ind” primitive, and transaction is ended.

(e) When the PDU Acknowledgement sent by the remote station through the “TransferData.indication” in the LPCP is received before the resend timer expires, the LPP stops the resend timer started up in (b) or (c), and completes this transaction.

(2) Receive processing

(a) When the Invoke PDU is received through the “TransferData.indicaton” in the LPCP, the LPP notifies the application of the received data through the “Invoke.ind” primitive.

(b) When the RA flag is valid in the PDU received in (a), the LPP prepares the PDU Acknowledgement, sends it to the remote station through the “TransferData.request” in the LPCP, and starts up the wait timer.

(c) When the Invoke PDU received in (a) is received again for some reason (such as that the PDU Acknowledgement sent in (b) does not arrive), the LPP aborts this PDU, prepares the PDU Acknowledgement again, sends it to the remote station through “TransferData.request” in the LPCP, and starts up the wait timer again.

(d) When the wait timer started up in (b) or (c) expires, the LPP ends this transaction is

ended.

Figure I5-6 shows an example of the sequence when resend processing is enabled. Figure I5-7 shows an example of the sequence when resend processing has succeeded. Figure I5-8 shows an example of the sequence when resend processing has failed.

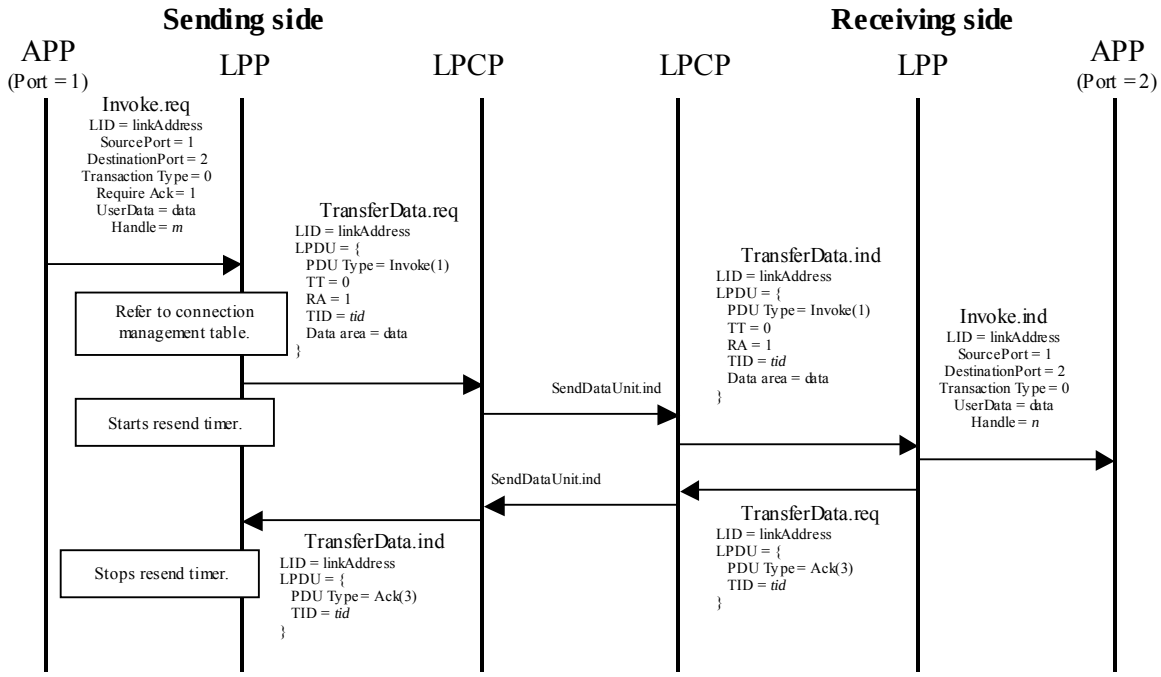


Figure I5-6 — Example of Resend Processing (Basic sequence)

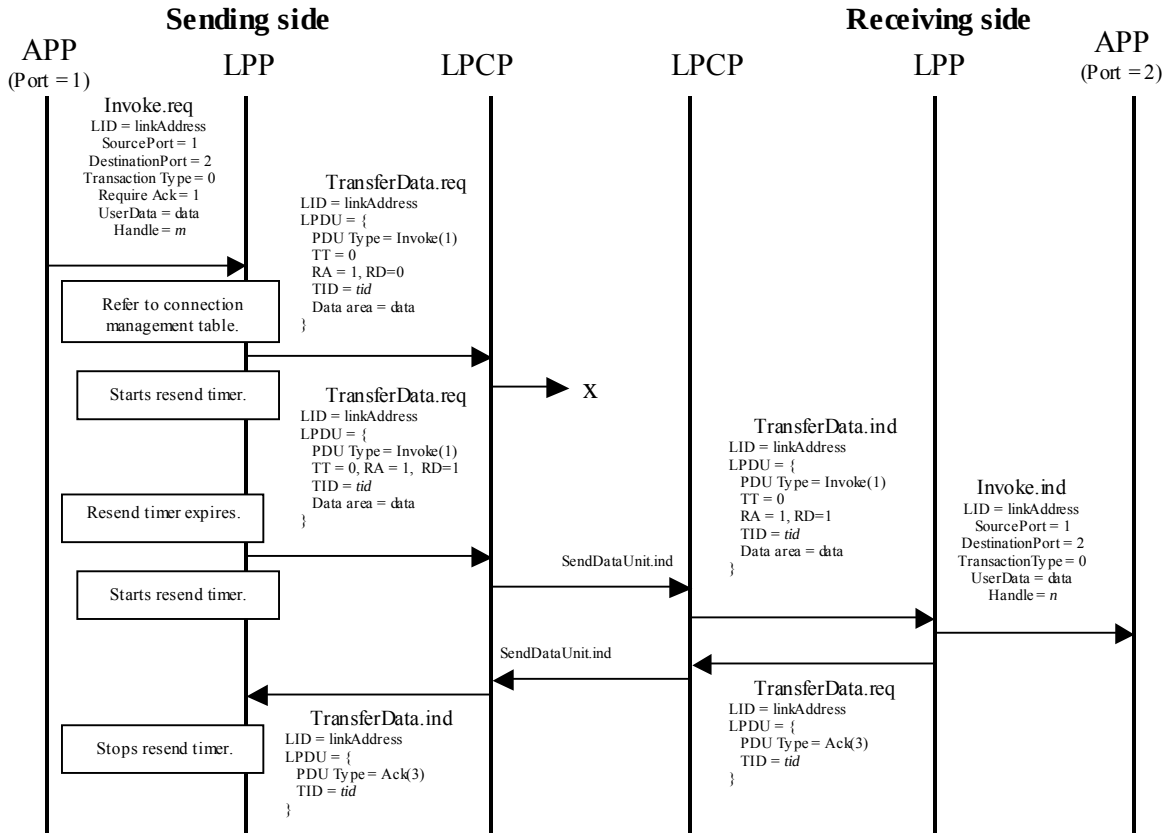


Figure I5-7 — Example of Resend Processing (When resend has succeeded)

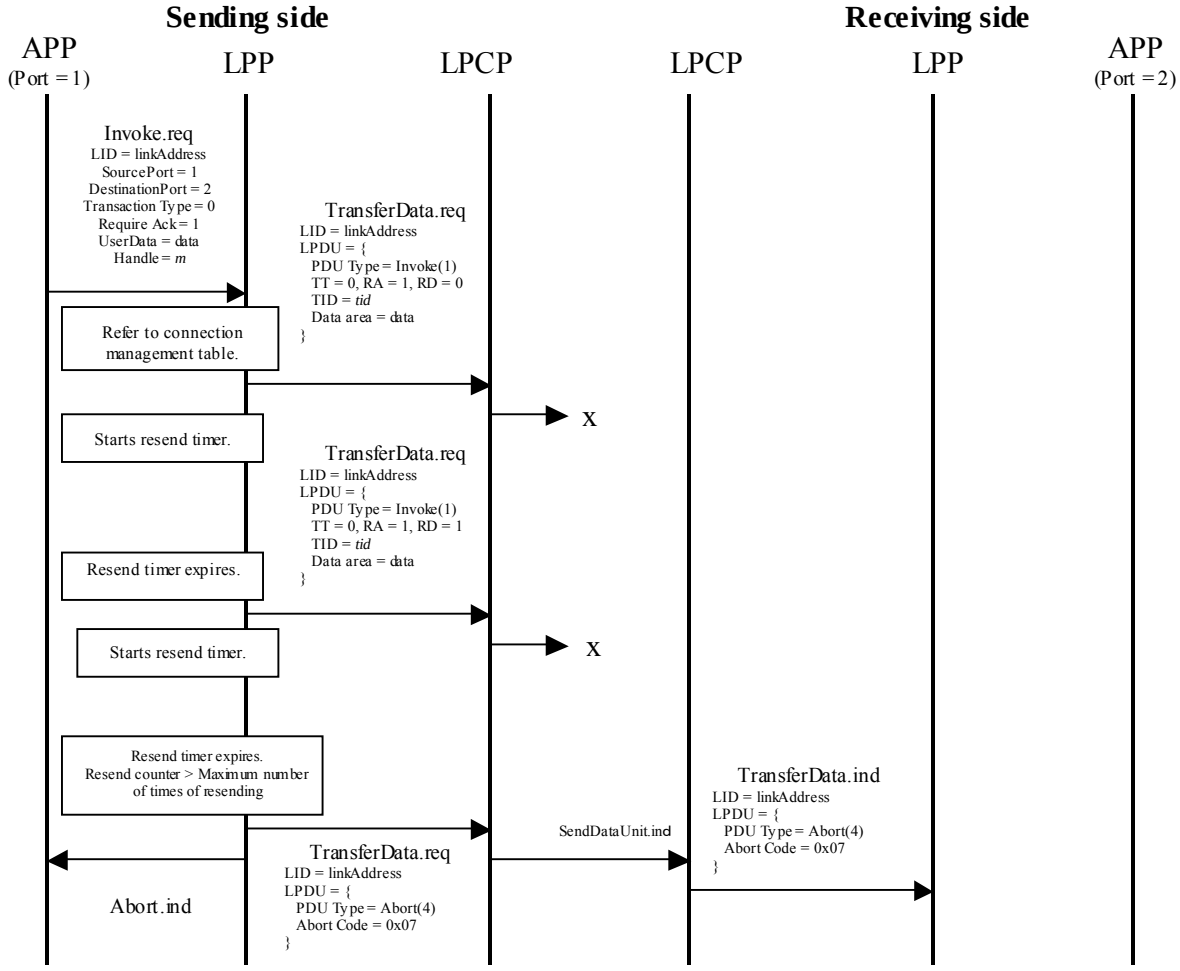


Figure I5-8 — Example of Resend Processing (When resend has failed)

I5.5 Segmentation/re-assembly Process (Optional)

Segmentation/re-assembly processing is used when a message exceeding the MTU in the LPCP is specified in “Invoke.req” and “Invoke.res”. This paragraph describes the sequence when segmentation/re-assembly processing is used in “Invoke.req”.

(1) Send processing

(a) When an application specifies a message whose size exceeds the MTU in the LPCP and issues the “Invoke.req” primitive, a transaction is started in the data sending service in which the segmentation/assembly processing is enabled.

(b) When the combination of specified link address and destination port number indicates a reject port, the LPP notifies the application of “Destination port error” through the “Abort.ind” primitive.

(c) When a transaction with the segmentation/assembly processing is already being executed for the combination of specified link address and destination port number, the

LPP notifies the application of "Under segmentation/assembly processing" through the "Abort.ind" primitive.

(d) In any case other than (b) and (c), the LPP disassembles the message from the top sequentially in the SUL, adds the header of the "InvokeSegment" PDU (described in I4.1.5) to each disassembled segment, and then send each segment in turn through the "TransferData.request" in the LPCP. At this time, the LPP assigns segment number (Segment No) "0" to the first segment, and then assigns an incremented value as the segment number in turn to each of the following segments. If the link address is a broadcast one and the Handle is same as that in the transaction with broadcast communication executed just before preceding the current transaction, the LPP sets the RD flag into each segment, and uses the same transaction ID in the transaction executed just before with the broadcast communication.

(e) When the sending queue in the ASL has overflowed and the LPCP notifies the LPP of the "The sending service is aborted because of sending queue overflow." through the "EventReport.indication", the LPP waits for a certain period of time, then starts sending again, including data whose sending failed previously.

(f) In the point to point communication, the LPP starts up the resend timer after sending the last segment data, waits to receive the PDU Acknowledgement or Nack from the remote station, and then performs the following processing. In the broadcast communication, the LPP completes the processing after sending the last segment data without performing the following processing.

(g) When the PDU Nack sent by the remote station through the "TransferData.indication" in the LPCP is received, the LPP resends the segments specified in Segment Number List in the PDU Nack. At this time, the LPP sets the RD flag to 1 in all segments to be sent again, and sets the FIN flag to 1 in the segment sent again at the end. After resending all segments, the LPP starts up the resend timer, and waits to receive the PDU Acknowledgement or Nack from the remote station.

(h) When the resend timer started up in (f) of (g) expires, the LPP sends the final segment again, and starts up the resend timer again.

(i) When the PDU Acknowledgement sent by the remote station through the "TransferData.indication" in the LPCP is received, the LPP stops the resend timer started up in (f), (g) or (h), and then completes this transaction.

(2) Receive processing

(a) An application specifies the buffer area for assembly of receiving data through the port registration primitive (RegisterPort).

(b) When the "InvokeSegment" PDU is received through the "TransferData.indication" in the LPCP, the LPP sets it based on sequence number in the buffer area.

(c) When the final segment is received, the LPP checks whether there are any un-received

segments. If there are un-received segments, the LPP prepares the PDU Nack, sends it to the remote station through the “TransferData.request” in the LPCP, and then memorizes the final segment number.

(d) When data whose RD flag is not set for some reason (such as a change of the arrival order after sending the PDU Nack in (c)) are received, the LPP aborts the data.

(e) In point to point communication, if the LPP has received all segment data when the final segment is received, the LPP notifies the application of the received data using the “Invoke.ind” primitive, prepares the PDU Acknowledgement, and sends it to the remote station using the “TransferData.request” in the LPCP.

(f) In the broadcast communication, when the LPP has received all segment data, the LPP notifies the application of the received data using the “Invoke.ind” primitive. If segment data included in this transaction are received after issuing the “Invoke.ind” primitive, the LPP aborts it.

Figure I5-9 shows an example of the basic sequence with the segmentation/assembly processing. Figure I5-10 shows an example of the sequence when some segment data are missing and selective resend processing is performed. Figure I5-11 shows an example of the sequence when the final segment data are missing and resend processing is performed. Figure I5-12 shows an example of the sequence for broadcast communication.

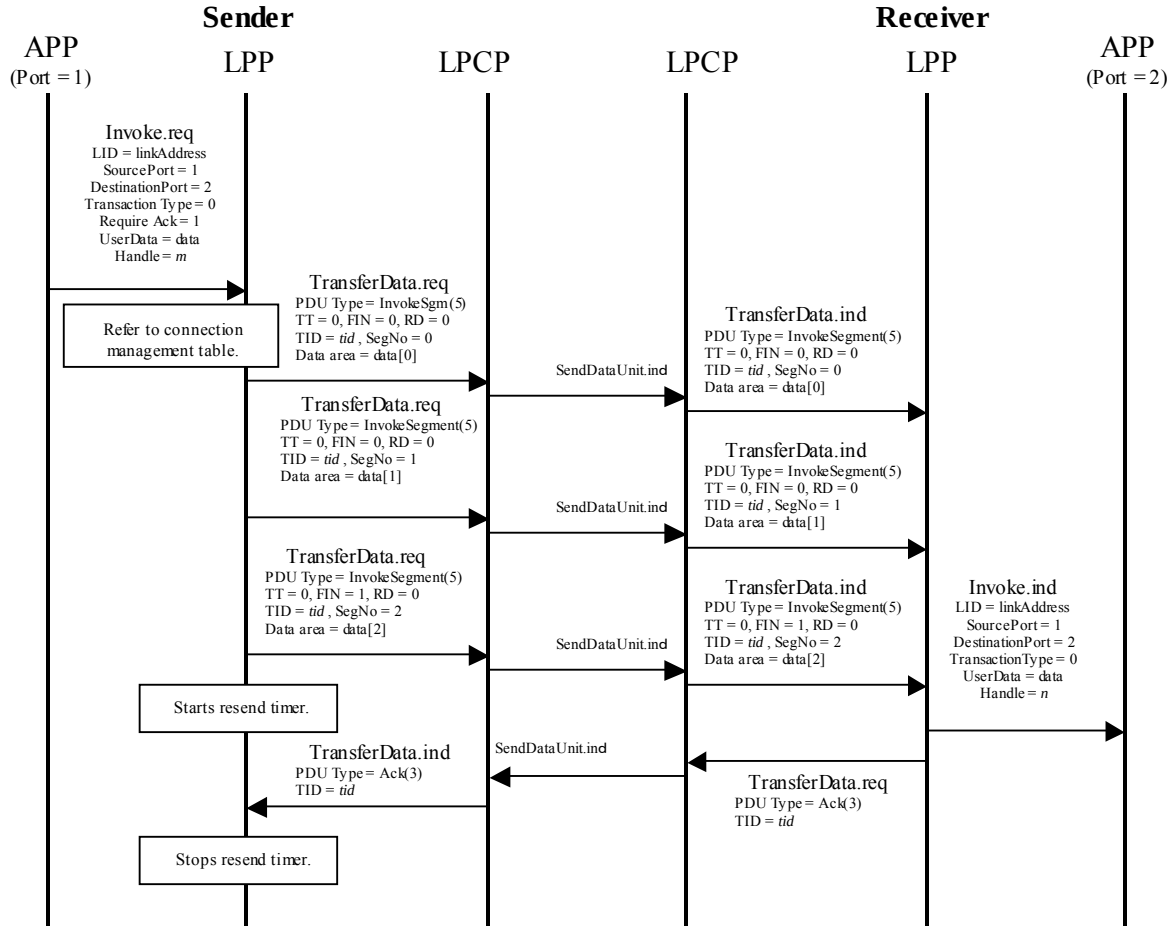


Figure I5-9 — Example of Segmentation/re-assembly processing

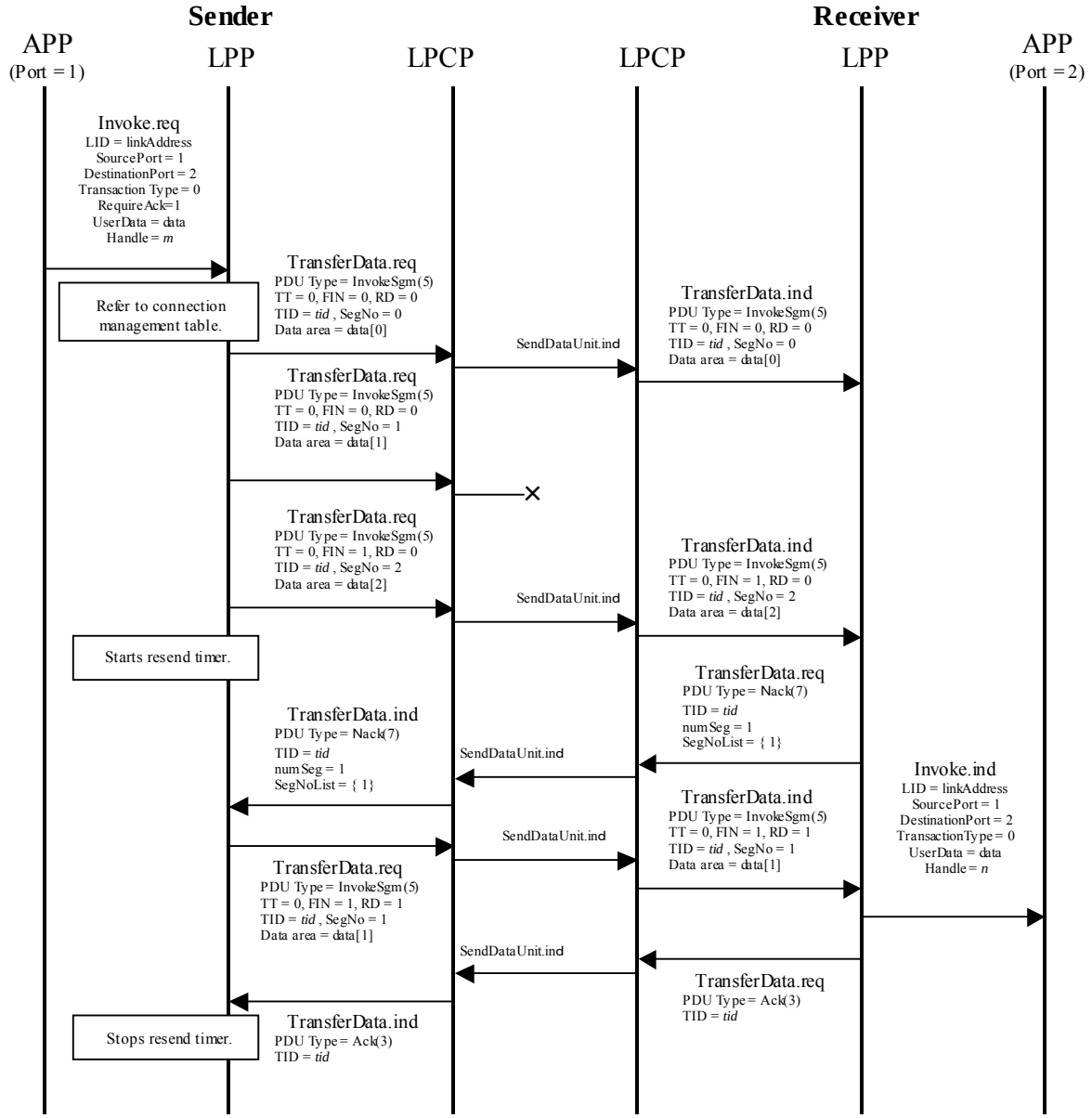


Figure I5-10 — Example of Segmentation/re-assembly Process (Selective Resend process)

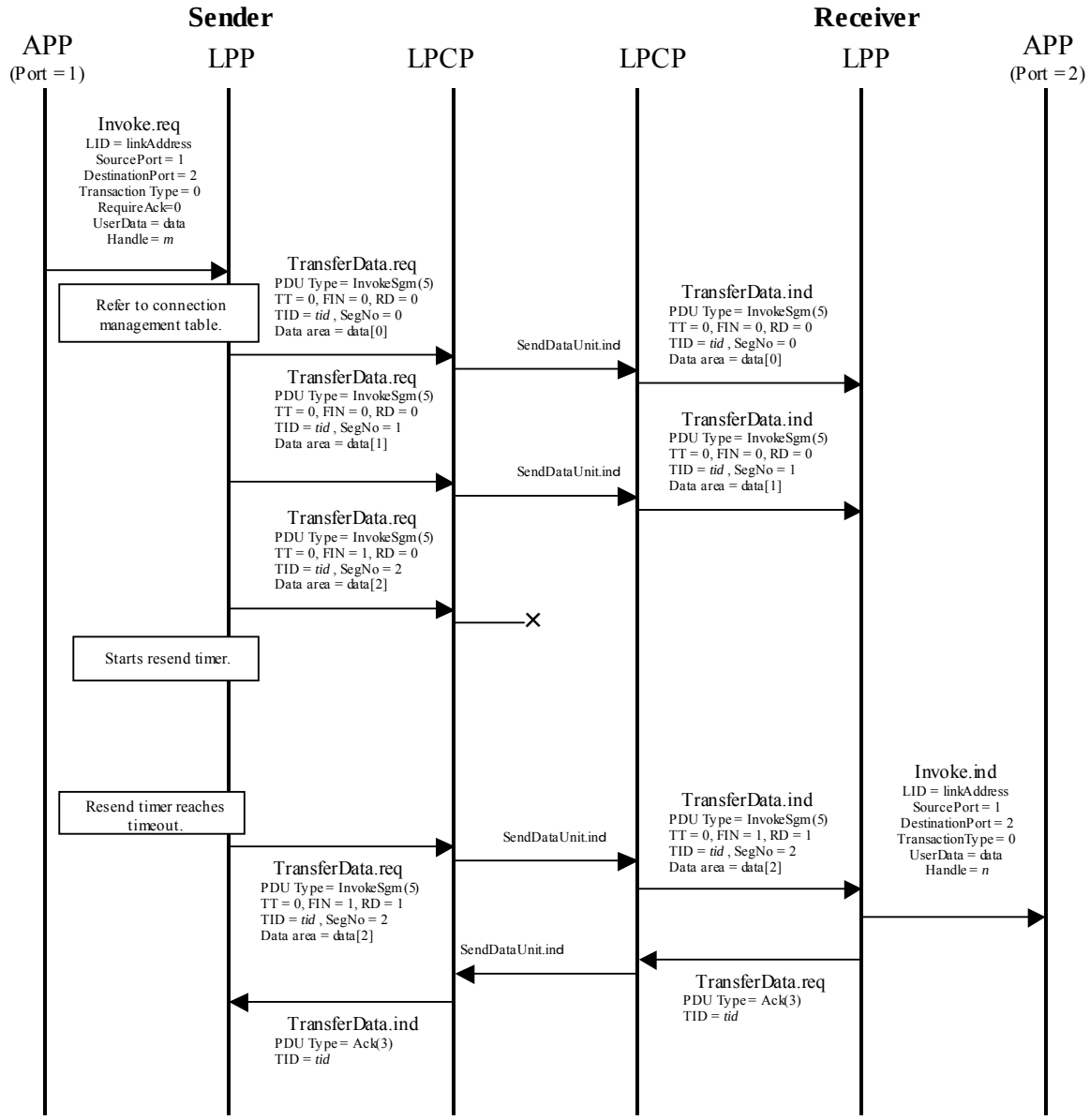


Figure I5-11 — Example of Segmentation/re-assembly Process (When final segment data have not been received)

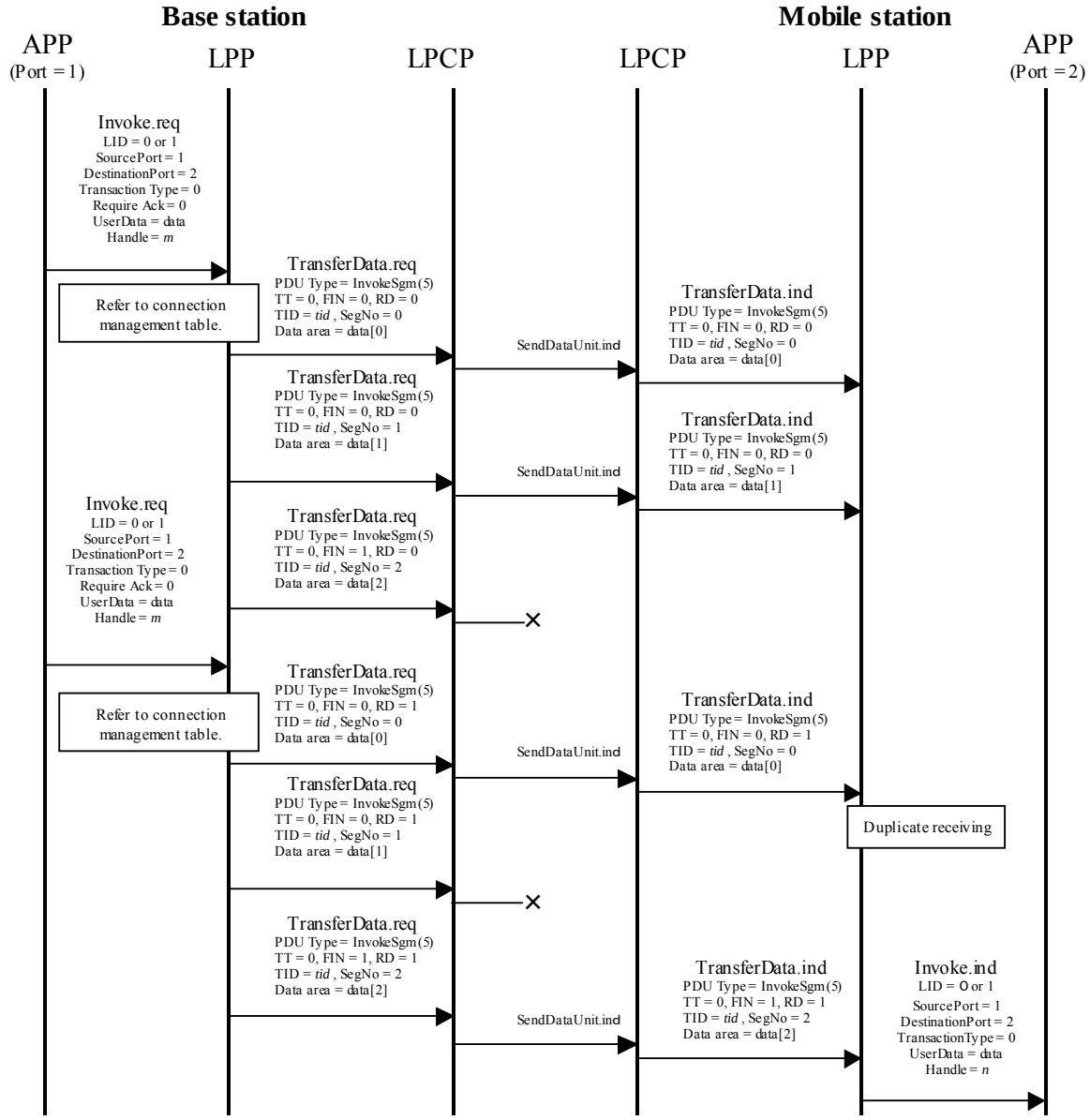


Figure I5-12 — Example of Segmentation/re-assembly process in Broadcast Communication

I5.6 Communication End Procedure

(1) Communication End Procedure

(a) The LPP receives the event “disconnection notice (97)” from the LPCP through the “EventReport.indication”.

(b) The LPP issues the “Disconnect.ind” primitive to the application, which is using the corresponding link address.

(c) The LPP deletes the connection management table for the link address received in the

primitive above. Thereafter, the LPP will not accept any transaction start each segment for this link address.

Figure I5-13 shows an example of the sequence in the communication end procedure.

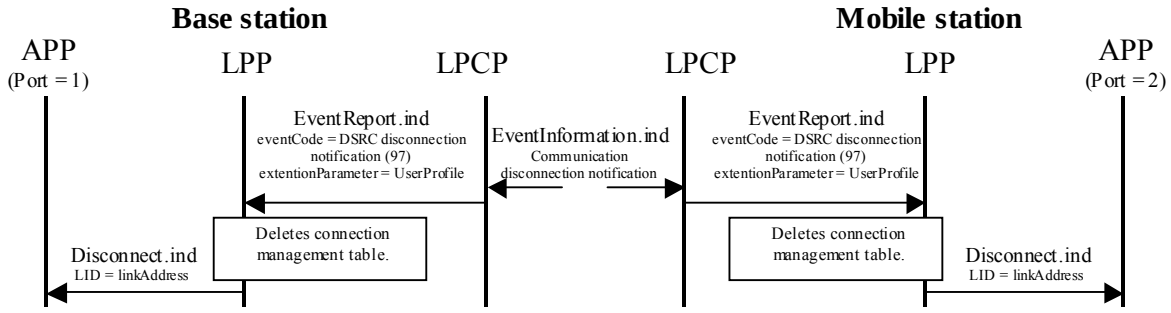


Figure I5-13 — Procedure when DSRC is Disconnected

I5.7 Transaction Abort Procedure

The LPP accepts a request from an application to abort a transaction when the transaction is in the following status:

(1) Sender side
From accepting “Invoke.req” to issuing “Invoke.cnf” in a request-response type transaction

(2) Receiver side
From issuing “Invoke.ind” to sending the Result PDU in a request-response type transaction

The sequence of transaction about procedure is as follows:

(a) When the LPP receives the “Abort.req” primitive from an application, this sequence is started.

(b) The LPP prepares the Abort PDU for the transaction specified in the primitive above, and then sends it to the remote station through the “TransferData.request” in the LPCP.

(c) The LPP issues the “Abort.ind” primitive to the requesting application to notify that aborting of the transaction has been completed.

(d) When the Abort PDU is received through the “TransferData.indication” in the LPCP, the LPP aborts all resources related to the transaction specified in the PDU above if it is being executed in the local station, and then issues the “Abort.ind” primitive to the application to notify that the transaction has been aborted.

Figure I5-14 shows an example of the sequence in the transaction abort procedure.

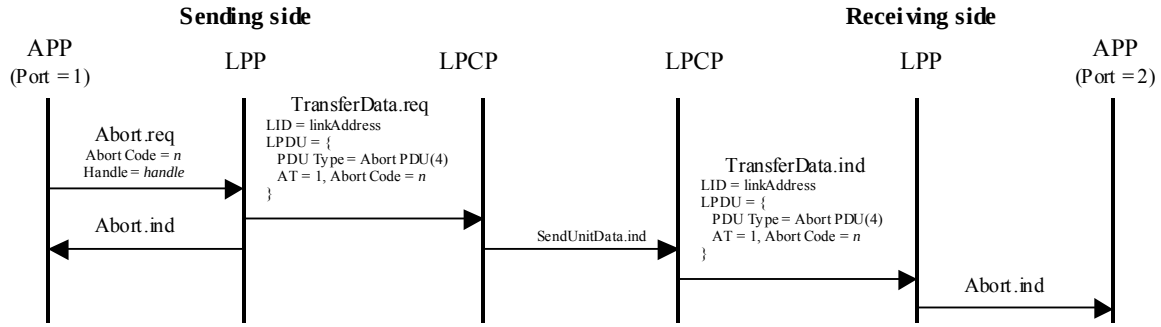


Figure I5-14 — Transaction Abort Procedure

I5.8 Accept Port Change Procedure

The LPP may dynamically change accept ports while the DSRC is connecting according to the registration and deregistration of accept ports by applications.

(1) Accept Port Addition Procedure

(a) While the DSRC is connecting, each application in a mobile station and the base station registers accept port numbers to the LPP using the port registration primitive (RegisterPort).

(b) The LPP updates the connection management table, and registers the accept port numbers specified in (a) as data receiving ports in the LPCP.

(c) The management service of the LPP prepares the PDU for notification of accept ports using the registered accept port numbers, and sends the PDU to the management service in the remote station using the “TransferData.request” in the LPCP.

(d) When the accept port notification sent in (c) is received, the management service of the LPP registers the received ports in the connection management table for the link address of this message. Thereafter, the LPP may accept transaction start requests to these accept ports.

Figure I5-15 shows an example of the sequence for the accept port addition procedure.

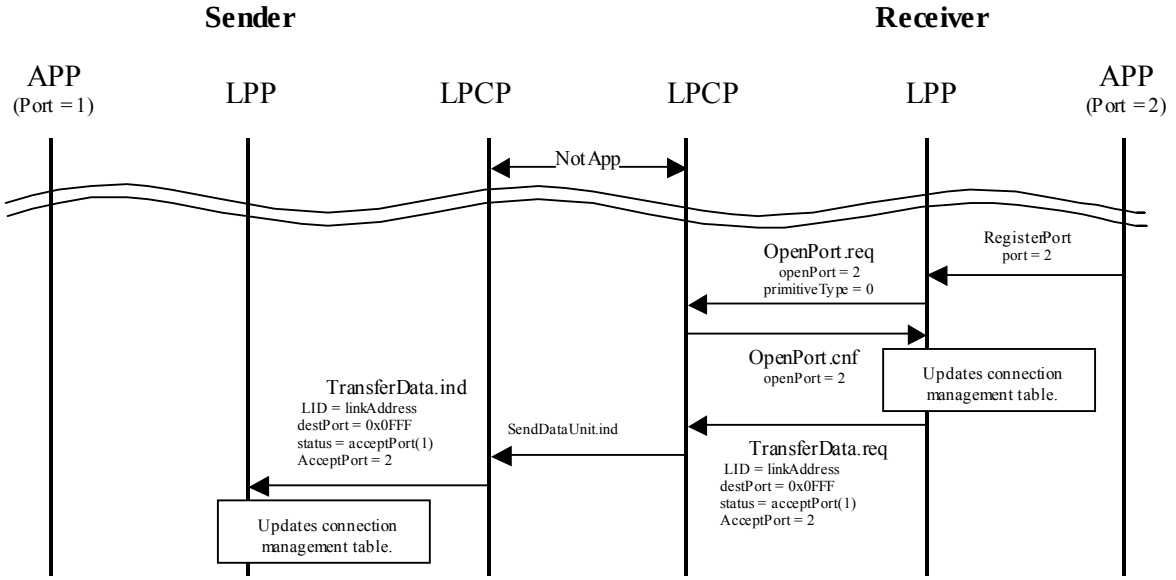


Figure I5-15 — Example of Accept Port Addition Procedure

(2) Accept port deletion procedure

(a) While the DSRC is connecting, each application in a mobile station and the base station deregisters accept port numbers from the LPP using the port deregistration primitive (DeregisterPort).

(b) The LPP updates the connection management table, and deregisters the accept port numbers deregistered in "0" from the accept port list in the LPCP using the "ClosePort.request".

(c) The management service of the LPP prepares the PDU for notification of reject ports using the deregistered port numbers, and sends the PDU to the management service in the remote station using the "TransferData.request" in the LPCP.

(d) When the reject port notification sent in "0" is received, the management service in the LPP deregisters the received ports from the connection management table for the link address of this message. Thereafter, the LPP will not accept transaction start requests to these reject ports.

Figure I5-16 shows an example of the sequence for the accept port deletion procedure.

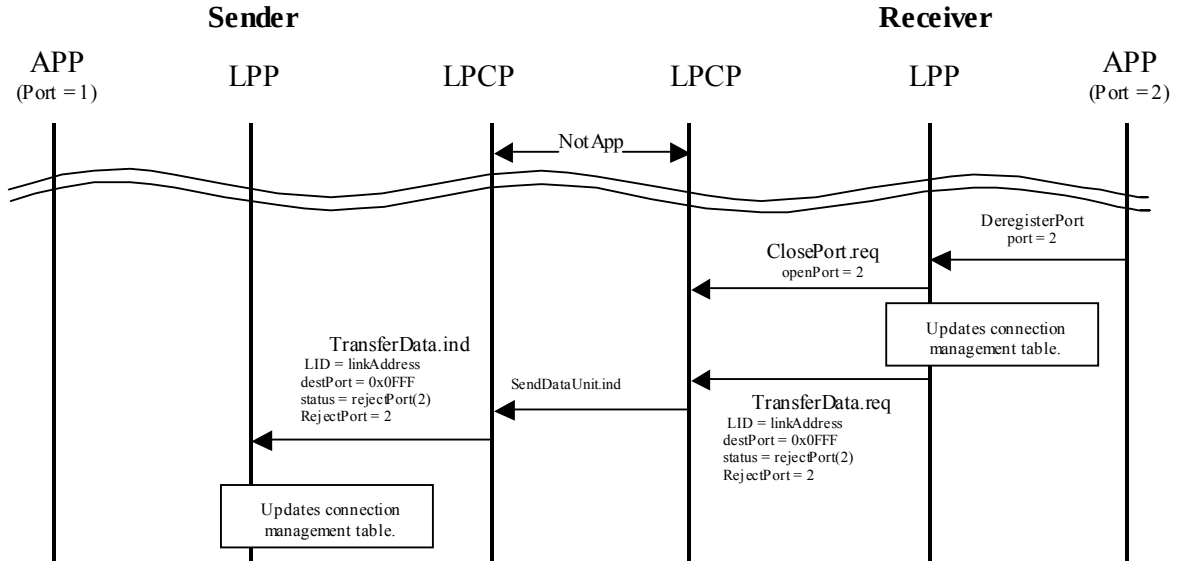


Figure I5-16 — Example of Accept Port Deletion Procedure

I6 Extension for Separated Type Configuration in Base Station

I6.1 Outline

The LPPoUDP is an extension protocol allowing the LPP in the external terminal to use the interface of the LPCP in the base station by using the datagram transfer service of UDP.

This extension protocol consists of the transfer service in the base station, the interface-providing entity in the external terminal and the event handling mechanism in the external terminal, and provides the following three types of interfaces which are provided to LPP by the LPCP in the base station to the LPP external terminal:

(1) TransferData.req
(2) TransferData.ind
(3) EventReport.ind

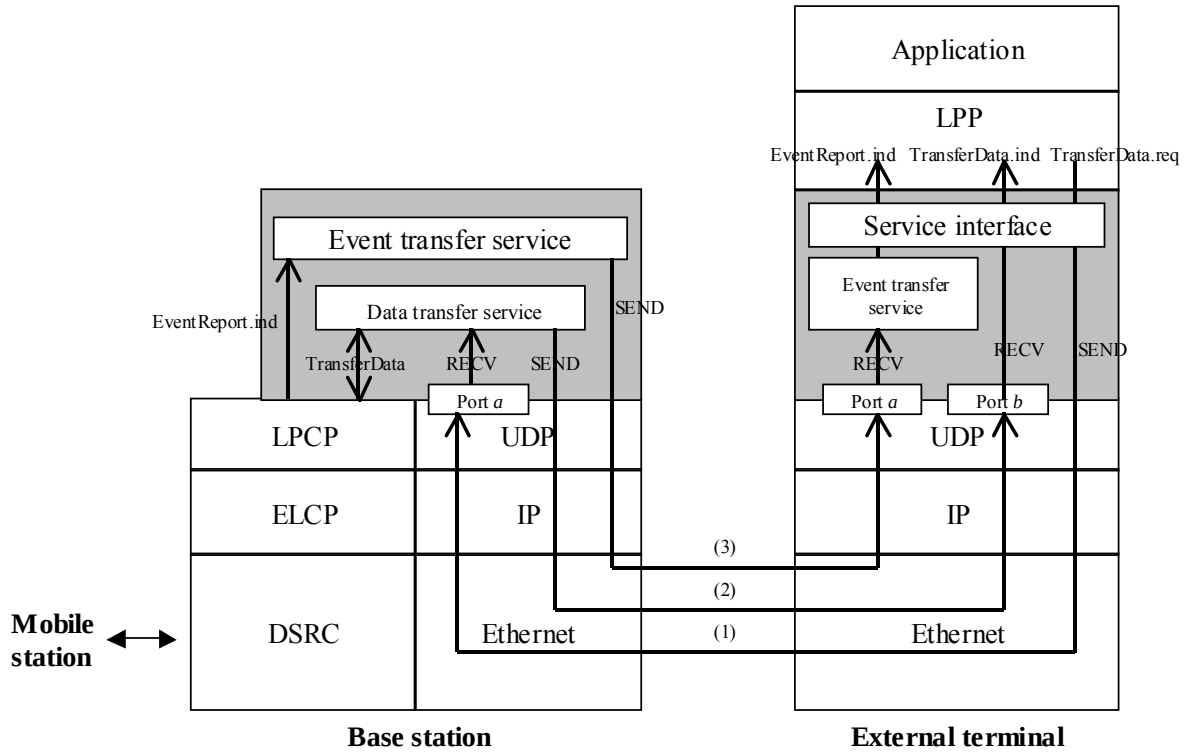


Figure I6-1 — Outline of LPP over UDP

I6.2 Definition of UDP and LPCP

Figure I6-2 and Figure I6-3 show the protocol data unit of the UDP and LPCP.

Figure I6-2 — UDP Datagram Format

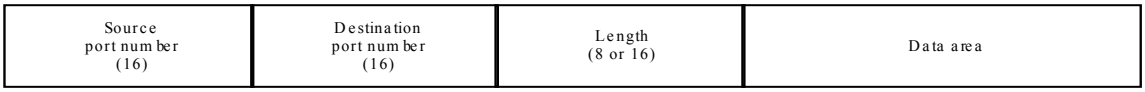


Figure I6-3 — PDU Format in LPCP

The notations for primitives of the LPCP and interfaces of the UDP are shown below:

(1) Primitives of the LPCP

TransferData.req (link address, source port number, destination port number, data sent)
TransferData.ind (link address, source port number, destination port number, data received)
EventReport.ind (status identifier, event additional information)

(2) Interfaces of the UDP
SEND (source IP address, sending destination IP address, source port No, destination port number, data sent)
RECV (source IP address, sending source port number, data received)

I6.3 Details of Mapping Method

I6.3.1 Mapping in Data Transfer Function (TransferData.req)

When the request "TransferData.req" is received from the LPP (implemented in the external terminal), the transfer service in the external terminal sends the request to the receiving port (Wellknown port) of the data transfer service in the base station by the SEND interface of the UDP. The data area in the UDP indicates the source port, destination port, link address and data area in the LPCP are stored in the format shown in Figure I6-4, and it is transferred to the ASL-NCP shown in Figure I6-5.

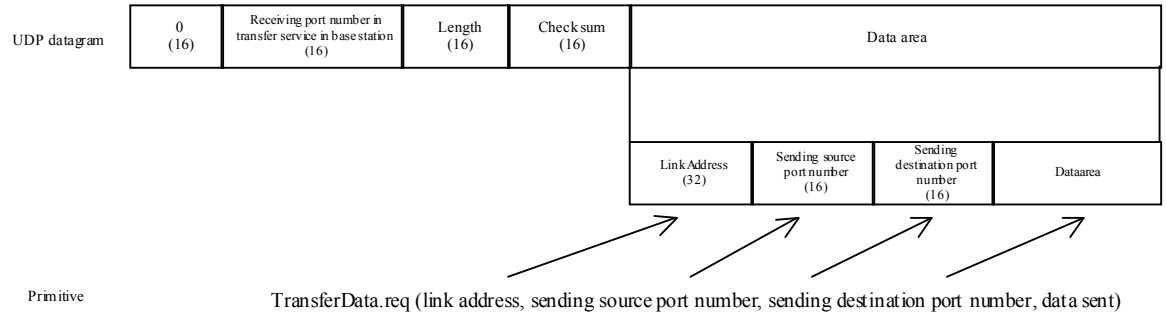


Figure I6-4 — PDU Mapping in TransferData.req

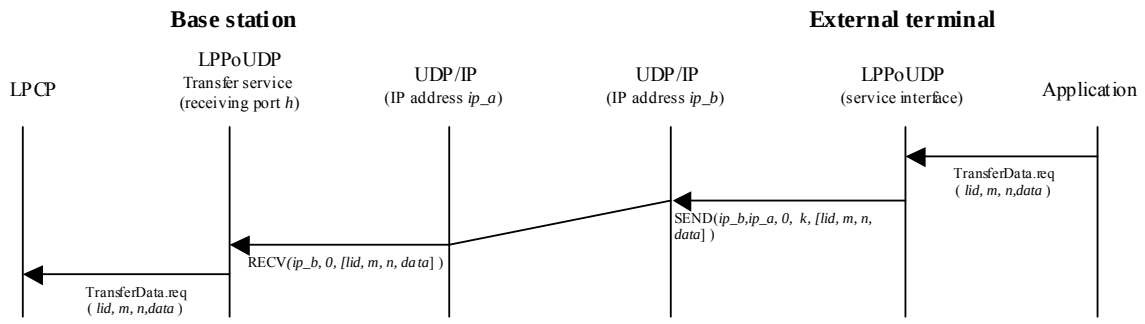


Figure I6-5 — TransferData.req Sequence

I6.3.2 Mapping in Data Transfer Function (TransferData.ind)

When “TransferData.ind” is received, the data transfer service in the base station sends it received message, whose destination source/port is set to the received LPCP source/port destination and to an external terminal by the SEND interface of the UDP source port. The destination port number in the LPCP is used to determine the sending destination external terminal. Accordingly, if there are two or more external terminals, port numbers in applications is unique in every external terminal. In the data area of the UDP indicates the link address and data area in the LPCP in the format shown in Figure I6-6, and it is notified of the LPP in the external terminal is stored shown in Figure I6-7.

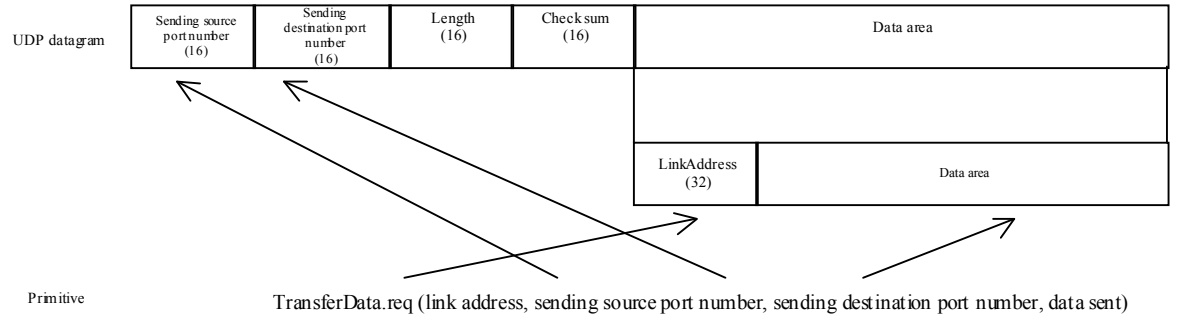


Figure I6-6 — PDU Mapping in TransferData.ind

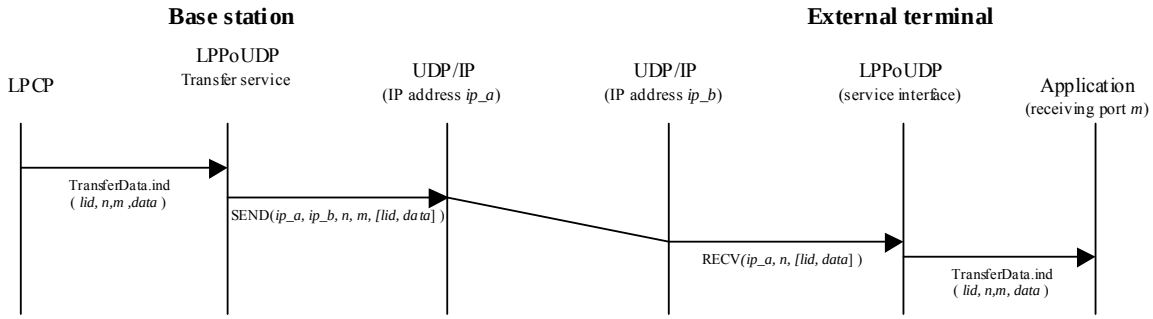


Figure I6-7 — TransferData.ind Sequence

I6.3.3 Mapping in Event Notification Function

The event transfer service works as an application with LPCP on the base station, and as an application with UDP on the external terminal. The information “EventReport” received by the event transfer service of the base station is sent to the event transfer service of the external terminal using the SEND interface of UDP. The receiving port number of the event notification service is 0x0ffe in the external terminal. In the data area of UDP, PDU indicate

the status identifier and additional event information are stored in the format shown in Figure I6-8, and it is notified of the LPP of the external terminal shown in Figure I6-9.

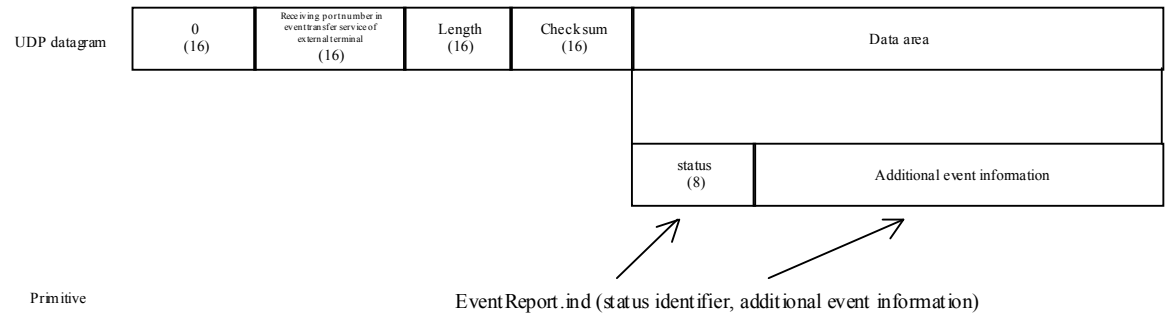


Figure I6-8 — PDU Mapping in EventReport.ind

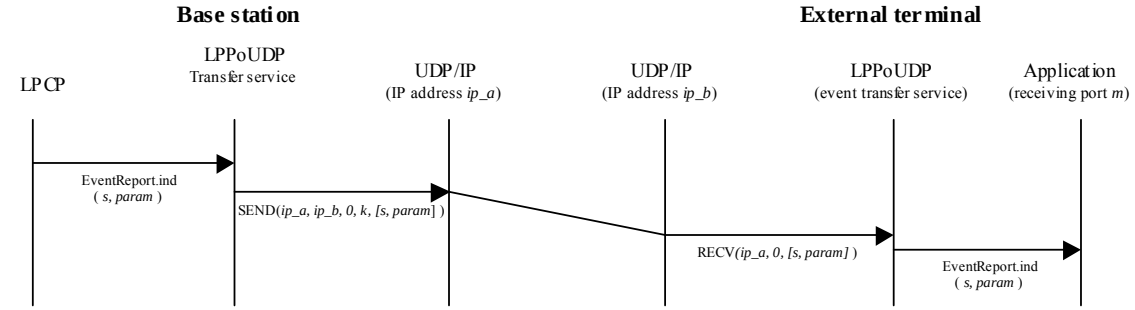


Figure I6-9 — EventReport.ind Sequence

I7 Definition of Parameter Types

```
LppParameter DEFINITIONS ::=
BEGIN

IMPORTS
PortNo     FROM     LocalControlParameter;

-- Transfer service processing
LppTransferDataPDU ::= CHOICE {
        notUse                  [0]     NULL,
        invokePdu               [1]     InvokePDU,
        resultPdu               [2]     ResultPDU,
        acknowledgementPdu      [3]     AcknowledgementPDU,
        abortPdu                [4]     AbortPDU,
        invokeSegmentPdu        [5]     InvokeSegmentPDU,
        resultSegmentPdu        [6]     ResultSegmentPDU,
        nackPdu                 [7]     NackPDU
        }
InvokePDU ::= SEQUENCE {
        version                 INTEGER(0..3),
        transactionType         INTEGER(0..1),
```

```
        requireAck          BOOLEAN,
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535),
        userData            OCTET STRING
        }
ResultPDU ::= SEQUENCE {
        fill                BIT STRING(SIZE(3)),
        requireAck          BOOLEAN,
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535),
        userData            OCTET STRING
        }
AcknowledgementPDU ::= SEQUENCE {
        fill                BIT STRING(SIZE(4)),
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535)
        }
AbortPDU ::= SEQUENCE {
        fill                BIT STRING(SIZE(4)),
        abortType           BOOLEAN,
        tid                 INTEGER(0..65535),
        abortCode           AbortCode
        }
InvokeSegmentPDU ::= SEQUENCE {
        version             INTEGER(0..3),
        transactionType     INTEGER(0..1),
        fin                 BOOLEAN,
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535),
        segmentNo           INTEGER(0..65535),
        userData            OCTET STRING
        }
ResultSegmentPDU ::= SEQUENCE {
        fill                BIT STRING(SIZE(3)),
        fin                 BOOLEAN,
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535),
        segmentNo           INTEGER(0..65535),
        userData            OCTET STRING
        }
NackPDU::= SEQUENCE {
        fill                BIT STRING(SIZE(4)),
        retransmitData      BOOLEAN,
        tid                 INTEGER(0..65535),
        segmentNumberList   SEQUENCE SIZE(0..65535) OF INTEGER(0..65535)
        }
AbortCode ::= INTEGER {
        unknown error                          (0),
        Protocol error                         (1),
        TID is invalid.                        (2),
        Transaction service is not supported. (3),
        LPP version is different.              (4),
        Receiving buffer has overflowed.       (5),
        MTU error                              (6),
        Resend timer timeout                   (7),
        Result time timeout                    (8),
        Link Address error                     (9),
        Destination port error                 (10),
        LPP is not supported.                  (11),
        Aborted by DSRC-ASL.                   (12),
        Transaction could not be started.      (13),
        Under segmentation/assembly processing(14)
```

```
        -- In AbortCode, values 15 to 255 are reserved.
        }(0..255)
```

-- Connection management service

LppStatusPDU ::= CHOICE {

```
        notUse      [0]      NULL,
        acceptPort [1]       AcceptPort,   -- Accept port
        rejectPort [2]       RejectPort,   -- Reject port
        dummy3 [3] NULL, dummy4 [4] NULL, dummy5 [5] NULL,
        dummy6 [6] NULL, dummy7 [7] NULL, dummy8 [8] NULL,
        dummy9 [9] NULL, dummy10 [10] NULL, dummy11 [11] NULL,
        dummy12 [12] NULL, dummy13 [13] NULL, dummy14 [14] NULL,
        dummy15 [15] NULL, dummy16 [16] NULL, dummy17 [17] NULL,
        dummy18 [18] NULL, dummy19 [19] NULL, dummy20 [20] NULL,
        dummy21 [21] NULL, dummy22 [22] NULL, dummy23 [23] NULL,
        dummy24 [24] NULL, dummy25 [25] NULL, dummy26 [26] NULL,
        dummy27 [27] NULL, dummy28 [28] NULL, dummy29 [29] NULL,
        dummy30 [30] NULL, dummy31 [31] NULL, dummy32 [32] NULL,
        dummy33 [33] NULL, dummy34 [34] NULL, dummy35 [35] NULL,
        dummy36 [36] NULL, dummy37 [37] NULL, dummy38 [38] NULL,
        dummy39 [39] NULL, dummy40 [40] NULL, dummy41 [41] NULL,
        dummy42 [42] NULL, dummy43 [43] NULL, dummy44 [44] NULL,
        dummy45 [45] NULL, dummy46 [46] NULL, dummy47 [47] NULL,
        dummy48 [48] NULL, dummy49 [49] NULL, dummy50 [50] NULL,
        dummy51 [51] NULL, dummy52 [52] NULL, dummy53 [53] NULL,
        dummy54 [54] NULL, dummy55 [55] NULL, dummy56 [56] NULL,
        dummy57 [57] NULL, dummy58 [58] NULL, dummy59 [59] NULL,
        dummy60 [60] NULL, dummy61 [61] NULL, dummy62 [62] NULL,
        dummy63 [63] NULL, dummy64 [64] NULL, dummy65 [65] NULL,
        dummy66 [66] NULL, dummy67 [67] NULL, dummy68 [68] NULL,
        dummy69 [69] NULL, dummy70 [70] NULL, dummy71 [71] NULL,
        dummy72 [72] NULL, dummy73 [73] NULL, dummy74 [74] NULL,
        dummy75 [75] NULL, dummy76 [76] NULL, dummy77 [77] NULL,
        dummy78 [78] NULL, dummy79 [79] NULL, dummy80 [80] NULL,
        dummy81 [81] NULL, dummy82 [82] NULL, dummy83 [83] NULL,
        dummy84 [84] NULL, dummy85 [85] NULL, dummy86 [86] NULL,
        dummy87 [87] NULL, dummy88 [88] NULL, dummy89 [89] NULL,
        dummy90 [90] NULL, dummy91 [91] NULL, dummy92 [92] NULL,
        dummy93 [93] NULL, dummy94 [94] NULL, dummy95 [95] NULL,
        dummy96 [96] NULL, dummy97 [97] NULL, dummy98 [98] NULL,
        dummy99 [99] NULL, dummy100 [100] NULL, dummy101 [101] NULL,
        dummy102 [102] NULL, dummy103 [103] NULL, dummy104 [104] NULL,
        dummy105 [105] NULL, dummy106 [106] NULL, dummy107 [107] NULL,
        dummy108 [108] NULL, dummy109 [109] NULL, dummy110 [110] NULL,
        dummy111 [111] NULL, dummy112 [112] NULL, dummy113 [113] NULL,
        dummy114 [114] NULL, dummy115 [115] NULL, dummy116 [116] NULL,
        dummy117 [117] NULL, dummy118 [118] NULL, dummy119 [119] NULL,
        dummy120 [120] NULL, dummy121 [121] NULL, dummy122 [122] NULL,
        dummy123 [123] NULL, dummy124 [124] NULL, dummy125 [125] NULL,
        dummy126 [126] NULL, dummy127 [127] NULL, dummy128 [128] NULL,
        dummy129 [129] NULL, dummy130 [130] NULL, dummy131 [131] NULL,
        dummy132 [132] NULL, dummy133 [133] NULL, dummy134 [134] NULL,
        dummy135 [135] NULL, dummy136 [136] NULL, dummy137 [137] NULL,
        dummy138 [138] NULL, dummy139 [139] NULL, dummy140 [140] NULL,
        dummy141 [141] NULL, dummy142 [142] NULL, dummy143 [143] NULL,
        dummy144 [144] NULL, dummy145 [145] NULL, dummy146 [146] NULL,
        dummy147 [147] NULL, dummy148 [148] NULL, dummy149 [149] NULL,
        dummy150 [150] NULL, dummy151 [151] NULL, dummy152 [152] NULL,
        dummy153 [153] NULL, dummy154 [154] NULL, dummy155 [155] NULL,
        dummy156 [156] NULL, dummy157 [157] NULL, dummy158 [158] NULL,
        dummy159 [159] NULL, dummy160 [160] NULL, dummy161 [161] NULL,
        dummy162 [162] NULL, dummy163 [163] NULL, dummy164 [164] NULL,
        dummy165 [165] NULL, dummy166 [166] NULL, dummy167 [167] NULL,
```

```
dummy168 [168] NULL, dummy169 [169] NULL, dummy170 [170] NULL,
dummy171 [171] NULL, dummy172 [172] NULL, dummy173 [173] NULL,
dummy174 [174] NULL, dummy175 [175] NULL, dummy176 [176] NULL,
dummy177 [177] NULL, dummy178 [178] NULL, dummy179 [179] NULL,
dummy180 [180] NULL, dummy181 [181] NULL, dummy182 [182] NULL,
dummy183 [183] NULL, dummy184 [184] NULL, dummy185 [185] NULL,
dummy186 [186] NULL, dummy187 [187] NULL, dummy188 [188] NULL,
dummy189 [189] NULL, dummy190 [190] NULL, dummy191 [191] NULL,
dummy192 [192] NULL, dummy193 [193] NULL, dummy194 [194] NULL,
dummy195 [195] NULL, dummy196 [196] NULL, dummy197 [197] NULL,
dummy198 [198] NULL, dummy199 [199] NULL, dummy200 [200] NULL,
dummy201 [201] NULL, dummy202 [202] NULL, dummy203 [203] NULL,
dummy204 [204] NULL, dummy205 [205] NULL, dummy206 [206] NULL,
dummy207 [207] NULL, dummy208 [208] NULL, dummy209 [209] NULL,
dummy210 [210] NULL, dummy211 [211] NULL, dummy212 [212] NULL,
dummy213 [213] NULL, dummy214 [214] NULL, dummy215 [215] NULL,
dummy216 [216] NULL, dummy217 [217] NULL, dummy218 [218] NULL,
dummy219 [219] NULL, dummy220 [220] NULL, dummy221 [221] NULL,
dummy222 [222] NULL, dummy223 [223] NULL, dummy224 [224] NULL,
dummy225 [225] NULL, dummy226 [226] NULL, dummy227 [227] NULL,
dummy228 [228] NULL, dummy229 [229] NULL, dummy230 [230] NULL,
dummy231 [231] NULL, dummy232 [232] NULL, dummy233 [233] NULL,
dummy234 [234] NULL, dummy235 [235] NULL, dummy236 [236] NULL,
dummy237 [237] NULL, dummy238 [238] NULL, dummy239 [239] NULL,
dummy240 [240] NULL, dummy241 [241] NULL, dummy242 [242] NULL,
dummy243 [243] NULL, dummy244 [244] NULL, dummy245 [245] NULL,
dummy246 [246] NULL, dummy247 [247] NULL, dummy248 [248] NULL,
dummy249 [249] NULL, dummy250 [250] NULL, dummy251 [251] NULL,
dummy252 [252] NULL, dummy253 [253] NULL, dummy254 [254] NULL,
dummy255 [255] NULL
-- In PDU LppStatus, identifier and type definition of tag numbers 3 to 255 are
-- reserved.
}

AcceptPort ::= PortNo

RejectPort ::= PortNo

END
```

Annex J
(informative)
Non-Network Application without using DSRC-ASL

J1 Overview

In an application (mainly network system) supposed by AID=18, the transaction between applications is done using DSRC-ASL specified in this standard. However, in other application (mainly non-network system) except for AID=18, it is investigated a method for the application to connect with DSRC protocol stack layer 7 (following DSRC-L7) directly like Electric Toll Collection System (ETC, AID=14), and a method an application sub layer (AHS-ASL) to intermediate between original application (AID=17) and DSRC-L7 same as architecture of DSRC-ASL of this standard is made, and to do transaction.

Interface specifications with ETC (AID=14) application and DSRC-L7 of ETC are shown as Japan Highway Public Corporation in accordance with ISO/TS 14906 specifications. In addition, interface specifications with DSRC-L7 in Network application (AID=18) mainly are specified in this standard in detail, too.

To utilize a DSRC protocol stack effectively, an application interface with DSRC-L7 investigated in AID=1 and AID=17 is described for reference.

In addition, there is not with the things that security does not mutual compatibility confirmation to be able to put between mobile station of an application identified in AID=1 or AID=17 in this Annex and application identified with this standard (AID=18) or AID=14, compatibility confirmation of mobile station is done security in each application identified with AID=1 or AID=17.

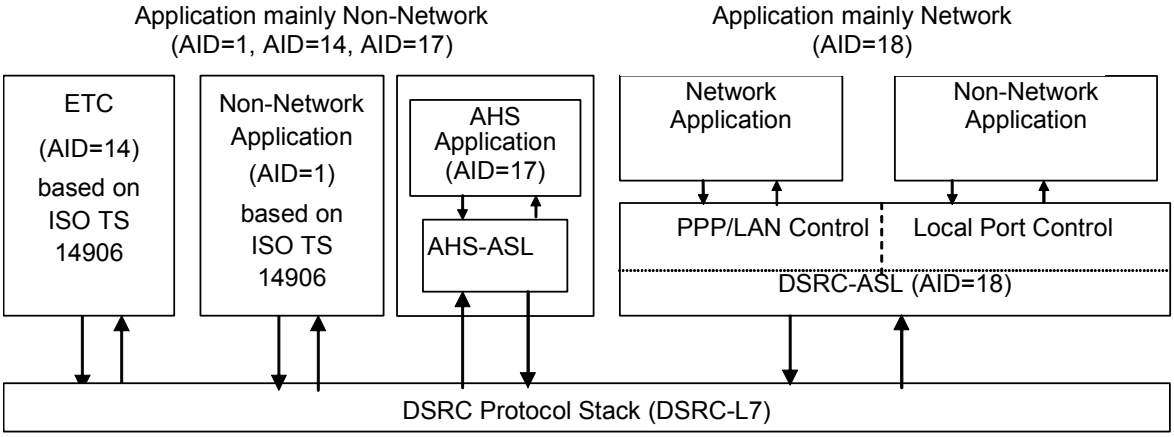


Figure J1-1 — Overview of Application Interface over DSRC Protocol Stack

J2 Non-Network Application Interface Identified by AID=1

In Annex I of the standard ARIB STD-T75, for example of association operation of application identified with AID of 14 based on ISO/TS 14906 “EFC Application Interface for DSRC”, the mechanism is explained that an application identified with unique Context between based station and mobile station which are implemented multi Context is selected. This annex defines as the identification procedure and the association procedure of the Non-Network application by the ContextMark in the AID of 1 defined ISO/TS 14906 simultaneously as a reference.

The application identified with AID of 1 as AID of 14 is an application refer to ISO/TS 14906 for an application interface specification, and is supposed to be adapted to Non-Network application using AID of 1 except ETC (AID=14) for an account settlement of payment mainly.

J2.1 Identification Procedure of Applications between Base Station and Mobile Station

In an individual communication using DSRC as a communication media, the base station and the mobile station should realize the initialization called the association at first of the communication.

The base station informs the mobile station of the Beacon Service Table (BST). The mobile station should reply the Vehicle Service Table (VST) in response to receive the BST.

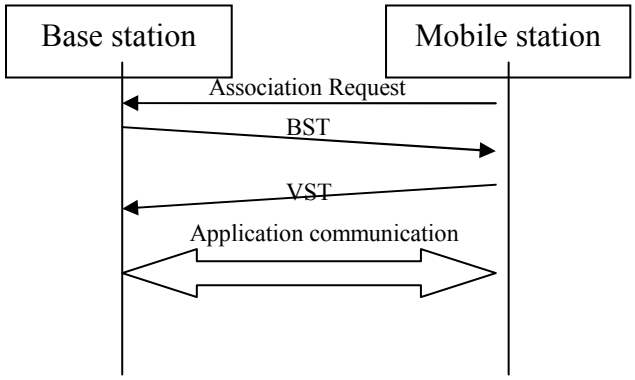


Figure J2-1 — Identification of Applications

The BST consist of the information of the application which the base station should provide,

and the VST consist of the information of the application that the mobile station wants to be serviced, these information are called the application list. The application list is the list that contains the information of the application registered in the each list, and it may contain plural applications.

The procedure of selecting application is as follows:

(1) The mobile station evaluates a received BST and selects requesting application in the application list supposed by the base station.

In case of the mobile station supposing the application of the base station, the mobile station reports the application of requesting execution to the base station and send the corresponding VST of the mobile station to the base station.

On the other hand, in case of the mobile station not supporting the application supported by the base station, after this occasion the mobile station should not exchange any information with the base station.

(2) The mobile station registers the application wanted to give a service in L7 (I Kernel element) of DSRC beforehand, and evaluates it when the base station receives the VST.

On the other hand, in case of the base station not supporting the application wanted by the mobile station, after this occasion the base station should not exchange any information with the mobile station.

The application list in the VST consists of a combination of the following three items for each application.

DSRC Application ID (AID)
DSRC-EID (EID)
Application Context Mark (MPP-Context Mark)

In this case, the AID is the identifier that identifies applications, and is specified by the ISO/CD 15628 “DSRC Application Layer”. The EID is the ID that is numbered by the application of the mobile station; the DSRC may transfer the data without consciousness of the application. That is the address for the data transfer.

The MPP-Context Mark shows the structure of the data element used by the application, and is used with the EID in pairs.

J2.2 Exchange of BST/VST

J2.2.1 Contents of BST in Application

In the base station supposing the application, the I-KE should be provided with the following information concerning each application especially.

(1) Application Identifier (AID) is as follows.
(a) Electronic Fee Collection (EFC): AID=1
(b) Multi Purpose Payment (MPP): AID=14

(1) The BST may contain multiple applications.

(3) The description of the ASN.1 concerning the contents of the BST is as follows:

```
BST ::=SEQUENCE {
   beacon                          BeaconID,
   time                            Time,
   profile                         Profile,
   mandApplications                ApplicationList,
   nonmandApplications             ApplicationList OPTIONAL,
   profileList                     SEQUENCE(0..=127,...) OF Profile
   }

where :
ApplicationList ::=SEQUENCE (0..127,...) OF
   SEQUENCE {
   aid              DSRCApplicationEntityID,     --aid=e.g.14
   eid              Dsrc-EID      OPTIONAL,      --empty
   parameter         Container     OPTIONAL
   }
```

Figure J2-2 — Structure of BST

J2.2.2 Contents of VST in Application

The following explanation is as the example in case of the MPP (AID=14), it is same as in case of EFC (AID=1). Each application and the corresponding contract should be in connection with MPP-ContextMark, as the following definition. In case of a mobile station supposing multiple MPP application, the order of the MPP-ContextMark contained in the VST should be necessary to correspond with the order of the user requirement. The base station should serve the first MPP-CotextMark in priority order in support of containing in the VST.

The description of the ASN.1 concerning the contents of the VST is as follows:

```
VST ::=SEQUENCE {
    fill                 Bit STRING (SIZE (4)),
    profile              Profile,
    applications         ApplicationList,
    obeConfiguration     OBEConfiguration
    }

where :
ApplicationList ::=SEQUENCE (0..127,...) OF
   SEQUENCE {
    aid          DSRCApplicationEntityID,     -- aid=e.g.14
    eid          Dsrc-EID        OPTIONAL,    -- eid=e.g. 2
    parameter    Container       OPTIONAL,    -- MPP-ContextMark
                                              -- plus any Information
  }
```

Figure J2-3 — Structure of VST

The mobile station supporting each application should provide the I-KE with the following information in connection with the application especially.

(1) AID is defined as follows:
   (a) EFC: AID=1
   (b) MPP: AID=14

(2) The value of EID should be unique in the mobile station over all session of the DSRC.
Also this value should be connected with the MPP-ContxtMark contained in the parameter logically.

(3) The parameter should be defined as the OCTET STRING of the Container CHOICE type, and should contain the MPP-ContextMark defined below. Additionally it may consist in addition of the information needed for the application.

(4) The MPP-ContextMark consists of a contract provider, a kind of a contract, and a version of a Context, and shows the particular Context of the mobile station.

The description of the ASN.1 concerning the contents of the MPP-ContextMark is as follows:

```
MPP-ContextMark ::= SEQUENCE {
    contractProvider   Provider,
    typeOfContract     OCTET STRING （SIZE(2)） ,
    contextVersion     INTEGER(0..127,..)
    }

Provider ::= SEQUENCE {
     countryCode            CountryCode,
     providerIdentifier IssuerIdentifier
   }

CountryCode::=BIT STRING (SIZE(10))
IssuerIdentifier::=INTEGER(0..16383)
```

Figure J2-4 — Structure of MPP-ContextMark

J2.3 Application Identification by the ContextMark

The application of the MPP (AID=14) and EFC (AID=1) is described above.

The application is identified with the ContextMark, concretely it is identified with the ContractProvider in the MPP-ContextMark.

The ContractProvider consists of the identification data of 24 bits, and its upper 10 bits are the CountoryCode, the remaining 14 bits are the ProviderID.

The CountoryCode is defined as ISO3166, and in Japan it is defined “0101110110” fixedly, so it is identified with the ProviderID actually. In case of the MPP/EFC the ProviderID is assigned the number from 0 to 1023 at present, and the ProviderID may be assigned the number from 1024 to 16383 upper than that number in case of the Non-Network application as a Parking Garage except the ETC, a Gas Station, etc.

The identification image using ContextMark is shown in Figure J2-5, and the example of the identification by the ProviderID is shown in Table J2-1.

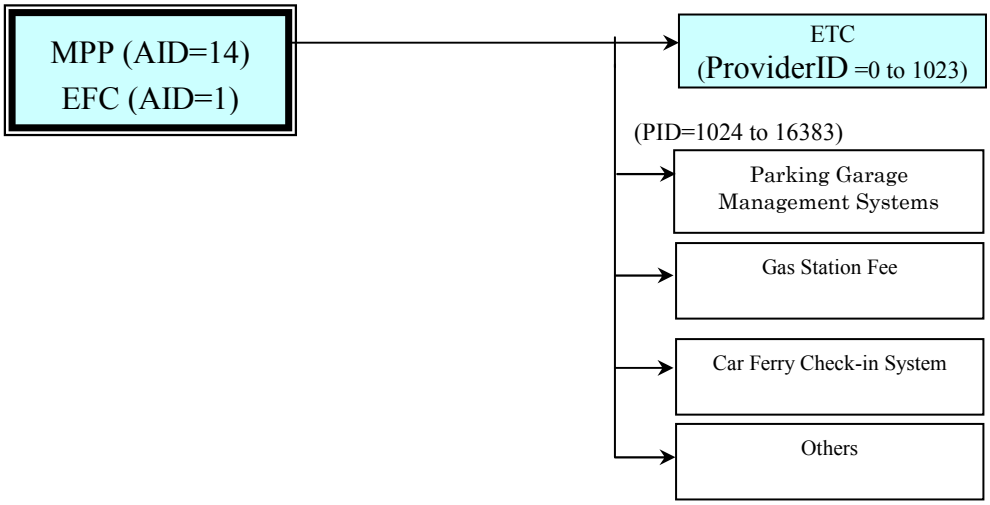


Figure J2-5 — Identification Concept of Non-IP Application using Context Mark

Table J2-1 — Example of Identification with Non-Network Application by ProviderID

Application		ProviderID		NOTE
		Upper 4bits	Lower 10 bits	
Fee Collection Application	ETC	0	0	Undefined
			1 to 1023	Defined as assignment (AID=14)
	Parking Garage Management Systems	1	0 to 1023	example
		2	0 to 1023	example
	Gas Station Fee	3	0 to 1023	example
		4	0 to 1023	example
	Car Ferry Check-in System	5	0 to 1023	example
	Spare	6 to 10	0 to 1023	example
Other Application	Physical Distribution Management	11	0 to 1023	example
	Commercial Car Management	12	0 to 1023	example
	Public Transportation Vehicle Management	13	0 to 1023	example
	reserved	14 and 15	0 to 1023	example

J3 Application Interface for the Cruse Assist Highway System Identified with AID of 17

(See Annex J2 of ARIB STD-T75 Japanese edition.)

Annex K
(informative)
Attention for Point to Point Protocol (PPP)

About the PPP that PPPCP handles, a matter about connectivity with base station and mobile station is described below.

K1 Authentication Protocol

Authentication protocols by PPP are such as non-authentication, PAP (Password Authentication Protocol), CHAP (Challenge-Handshake Authentication Protocol). For PPP connection the appropriate authentication protocol should be selected from these in base station and mobile station. Concretely, it is considered that non-authentication protocol in case of no needing authentication, or the PAP in case of authenticating considering processing time, or the CHAP in case of needing the password protection should be selected.

K2 DNS Server Address Setting

It is desirable for a DNS server address of equipment in the vehicle to set by means of RFC1877 (PPP Internet Protocol Extensions for Name Server Address) automatically. However, it is not limited when automatic setting of a DNS server address is unnecessary.

K3 Maximum Receive Unit (MRU) Option Setting

It is recommended that the PPP stack of a mobile station should use the stack that is negotiable to set the MRU to 1492 by the MRU optional setting of RFC1661 designated by the base station. In case of using the PPP stack that doesn't accept this setting, the MTU size of the PPP of a mobile station should be less than 1492 beforehand if necessary.

NOTE In case of using the protocol that set the MRU size of the base station less than 1500 octets in the base station inside, for example it is consideration that the mobile station should correspond to the PPP over Ethernet.

K4 Async-Control-Character-Map (ACCM) Option Setting

In order to secure a communication band by not transmitting unnecessary synchronous control character, it is desirable for a value of Async-Control-Character-Map (ACCM) to set "0".

DSRC APPLICATION SUB-LAYER

ARIB STANDARD

ARIB STD-T88 VERSION 1.0

Version 1.0 May 2004

Published by
Association of Radio Industries and Businesses

Nittochi Bldg. 11F
1-4-1 Kasumigaseki, Chiyoda-ku, Tokyo 100-0013, Japan
TEL 81-3-5510-8590
FAX 81-3-3592-1103

Printed in Japan